ADA 069288

AFFDL-TR-78-136

A USER'S MANUAL FOR A COMPUTER PROGRAM TO GENERATE FATIGUE SPECTRUM LOADING SEQUENCES

McDonnell Douglas Corporation
Douglas Aircraft Company
3855 Lakewood Blvd.
Long Beach, California 90846

20080815 141

November 1978

TECHNICAL REPORT AFFDL-TR-78-136
Final Report for Period September 1976 - September 1978

AIR FORCE FLIGHT DYNAMICS LABORATORY
AIR FORCE WRIGHT AERONAUTICAL LABORATORIES
AIR FORCE SYSTEMS COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OHIO 45433

NOTICE

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

John M. Potter

JOHN M. POTTER,
Project Engineer

J Olsen

JAMES J. OLSEN, Acting Chief
Structural Integrity Branch

FOR THE COMMANDER

Ralph L Kuster Jr

RALPH L. KUSTER, JR., Col, USAF
Chief, Structures & Dynamics Division

"If your address has changed, if you wish to be removed from our mailing list, or if the addressee is no longer employed by your organization please notify AFFDL/FBE, W-PAFB, OH 45433 to help us maintain a current mailing list".

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

AIR FORCE/56780/16 April 1979 — 350

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER AFFDL-TR-78-136	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A User's Manual for a Computer Program to Generate Fatigue Spectrum Loading Sequences		5. TYPE OF REPORT & PERIOD COVERED Final Technical Report Sept 1976 - Sept 1978
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) P. R. Abelkis P. M. Lee B. P. Tate		8. CONTRACT OR GRANT NUMBER(s) F33615-76-C-3116
9. PERFORMING ORGANIZATION NAME AND ADDRESS Douglas Aircraft Co. McDonnell Douglas Corp. 3855 Lakewood Blvd. Long Beach, Calif. 90846		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Project 486U, Task 486U02, Work Unit 486U0222
11. CONTROLLING OFFICE NAME AND ADDRESS Air Force Flight Dynamics Laboratory (AFAL/FB) Air Force Wright Aeronautical Laboratories Wright-Patterson Air Force Base, Ohio 45433		12. REPORT DATE November 1978
		13. NUMBER OF PAGES 249
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Computer Program
Spectrum Loading
Sequence Generation
Fatigue
Crack Growth

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The report contains, in the form of a user's manual, the listing and complete description of a computer program to generate fatigue spectrum loading sequences. The program is specifically tailored for the development of random cycle-by-cycle, flight-by-flight loading sequences typical of aircraft structures. However, its general features allow the development of any type spectrum. The random sequence of cycles and flights is produced by a random number generator. Alternate non-random flight sequences can

also be generated. The basic input data consists of loads exceedances spectra or data to calculate such spectra by the program. The program contains the following spectrum editing features: (1) truncation - elimination of cycles as a function of range and R, peak or valley, (2) clipping - loads below or above a specified clipping value are set equal to that value, (3) all loads in the spectrum are multiplied by a constant. The output is a valley, peak sequence of the loads spectrum. The spectrum may be read into a magnetic tape to be used in other analyses or in testing.

ADA069288

FOREWORD

This report was prepared by Douglas Aircraft Company, Long Beach, California for the Structural Integrity Branch, Structural Mechanics Division, Air Force Flight Dynamics Laboratory, Wright-Patterson Air Force Base, Ohio. The report was prepared under Contract F33615-76-C-3116, Project 486U02 "Advanced Metallic Structures," Work Unit 486U0222, "Effect of Transport/ Bomber Spectrum on Crack Growth." The project technical monitor was Mr. J. M. Potter, AFFDL/FBE.

The program was developed in the Advanced Technology Section Fatigue and Fracture Mechanics and Structural Methods Groups of the Structures Engineering Subdivision under the leadership of Messrs. T. Swift, D. Smillie and M. Stone. The program manager was Mr. J. Palmer. Mr. P. R. Abelkis was the program technical director. Computer programming was performed by Ms. P. M. Lee and Mr. B. P. Tate with support from Mr. G. Agajanian of McAuto - Scientific Programming Group.

The report was released by the authors for publication in November 1978.

CONTENTS

SECTION I

INTRODUCTION

This report describes a computer program to generate fatigue spectrum loading sequences. The report contains all information needed for the practical use of the program.

The program is specifically tailored for the development of random cycle-by-cycle, flight-by-flight loading sequences typical of aircraft structures. However, it can be used to generate a spectrum of any type for any structure.

The random sequence of cycles and flights is produced by a random number generator. The basic loads occurrences spectrum is calculated through the use of the computer program A6PA (program 16PA in Reference 1). The description of this program is not repeated here, but is to be found in Reference 1, with the exception of changes described in Appendix A.

The program is written in Fortran IV language for CDC CYBER 74 computer or its equivalent. Douglas Aircraft Company program identification number is A6PD. The program was extensively used to generate loads spectra sequences in the Air Force study "Effect of Transport/Bomber Spectrum on Crack Propagation," Reference 2.

SECTION II

PROBLEM STATEMENT

Structures subjected to periodic or non-periodic fluctuating loadings experience gradual degradation in its load carrying capability through formation and propagation of fatigue cracks. Often, this type of loading is referred to as "fatigue loading". Fatigue loadings can be divided into two basic categories:

1. Constant amplitude loading - fatigue loading in which all of the peak loads (maximum loads) are equal and all of the valley loads (minimum loads) are equal,

2. Spectrum loading - fatigue loading in which all of the peak loads are not equal and/or all of the valley loads are not equal; this loading is also known as variable amplitude or irregular loading.

Fatigue loading basic terms and examples of spectrum loadings are illustrated in Figures 1 and 2.

Most fatigue loadings experienced by aircraft structures are of the spectrum type. The spectrum features are a function of aircraft type and utilization, loading environments and type of structure. The sequence of loading is usually of random nature and is a function of loading environments and sequence of missions and flights.

The basic loads spectra contents are usually supplied in the form of loads exceedances or occurrences information or other statistical data which do not supply any information about the sequence of these loads. An exceedances/occurrences spectrum specifies the number of times that the various loads in the spectrum are exceeded or occur in a specified period of time. An example of such information for Air Force aircraft is given in Reference 3.

In view of the fact that the fatigue failure process in crack initiation and propagation stages is loading sequence dependent, means are needed to generate representative fatigue loading sequences which can be used in design, analysis and verification testing. The computer program described herein provides the means of generating a random sequence spectrum from given exceedances/occurrences spectra, including the generation of flight random sequence from a specified mission-flight mix.

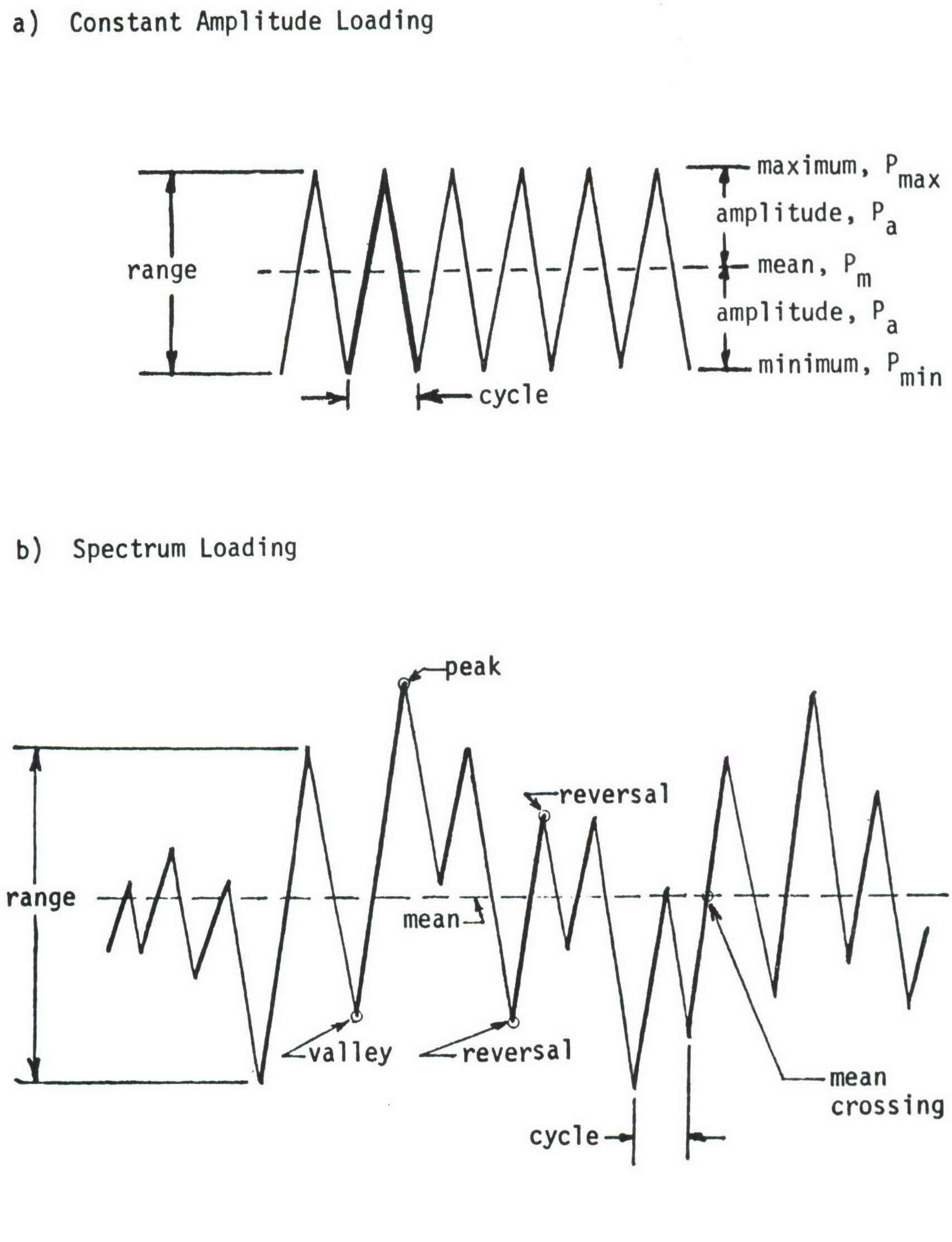


Figure 1. Fatigue Loading Basic Terms

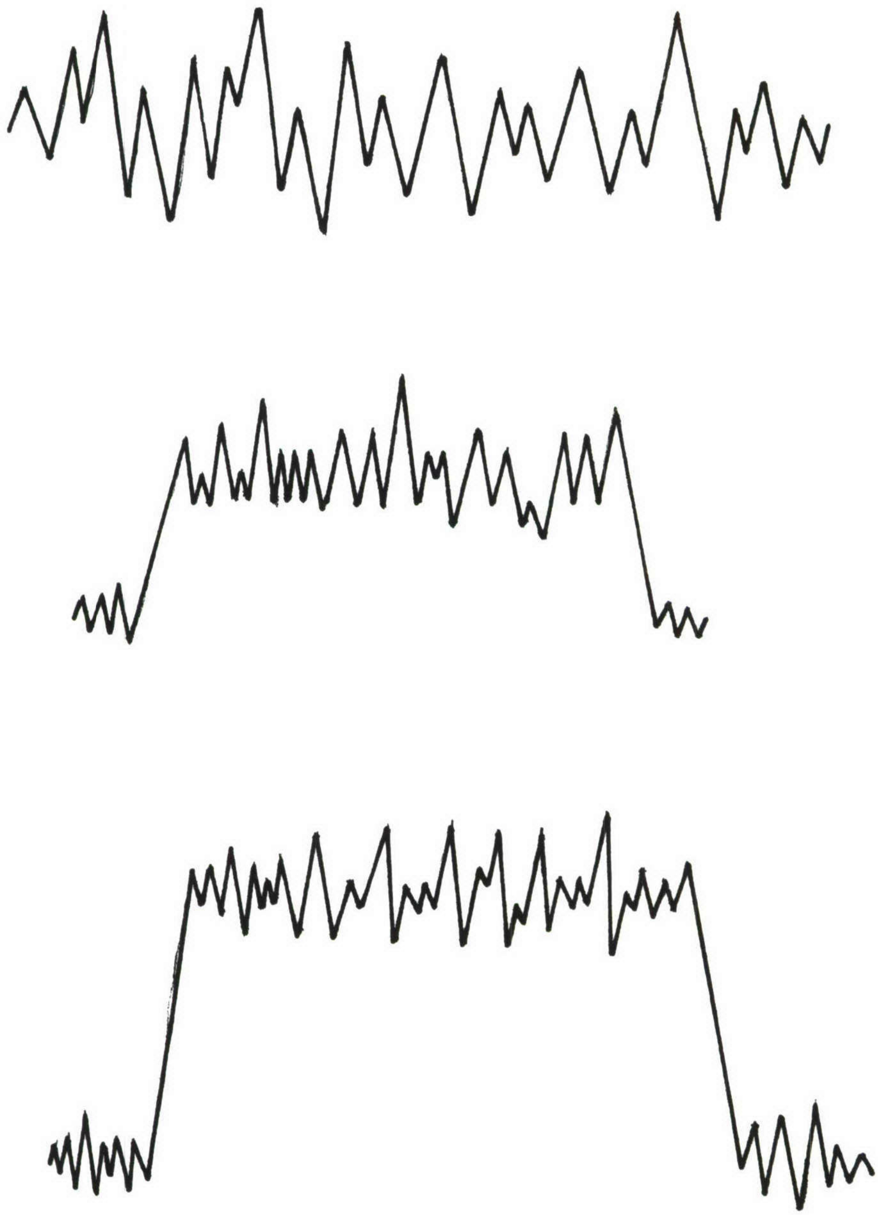


Figure 2. Examples of Spectrum Loadings.

SECTION III

PROGRAM OUTLINE

The program name is Spectrum Loading Sequence Generation (or RANDOM) and its identification is A6PD. It consists of program A6PA, described in Reference 1, and Appendix A, which generates the occurrences spectrum, and the additions to generate and edit random cycle-by-cycle loading sequences. A general outline of the program is given by Figure 3. A detail step-by-step explanation of the program, beyond the A6PA part, is given below.

1. The basic information extracted from A6PA, to generate the loading sequence, are the segment-by-segment occurrences spectra (MIN,MAX,n). However, program A6PD contains the entire A6PA program and executes all its features.

2. The occurrences spectrum generated by A6PA must be defined by the following A6PD input:

   NFT = Number of flight types. A flight type can be a realistic flight or any group of cycles to be identified as a "flight". (Card 2)

   NF = Number of flights for each flight type. (Card 4)

   NS = Number of A6PA segments in each flight type. (Card 5)

3. All A6PA segments in each flight type must be identified by input flags F1 and F2 . These flags are integers and must start with 1 and increase consecutively. Enter F1=F2=0 for any A6PA segment to be excluded. If the cycles (n) of several A6PA segments with the same MIN,MAX values are to be summed, enter the same F1 flag for these segments. If several A6PA segments with different MIN,MAX values are to be combined, enter the same F2 flag for those segments. A6PA segments are summed or combined, because in the loading sequence generation it is sometimes more realistic to treat several A6PA segments as one. Such situations might exist where several A6PA segments represent loadings in the same flight interval, such as maneuvers and gusts, but it is desired to have the cycles distributed within one F2 segment. In the sequence generation, the segment sequence is the F2 sequence. Example of combining A6PA segments under F1 and F2 flags is given on the following page.

A6PA Segm. No.	MIN	MAX	n	F1	F2	F2 Segm. MIN	F2 Segm. MAX	F2 Segm. n
1	8,000	12,000	100	1	1	8,000	12,000	125
	7,000	13,000	55			7,000	13,000	62
	6,000	14,000	12			6,000	14,000	17
	5,000	15,000	1			5,000	15,000	3
2	8,000	12,000	25	1	1	10,000	12,000	200
	7,000	13,000	7			10,000	13,000	100
	6,000	14,000	5			10,000	14,000	10
	5,000	15,000	2			10,000	15,000	2
3	10,000	12,000	200	2	1			
	10,000	13,000	100					
	10,000	14,000	10					
	10,000	15,000	2					

4. A6PA segment occurrences spectra are formed into a

$$(MIN, MAX, n, n/FLT)$$

array according to flight type and F2 segment. All cycles n are rounded off to whole numbers and $n<.5$ are set to zero.

If $\Sigma(n/Flt)$ for a given F2 segment is not an integer, a 'fictitious' load level

$$(MIN_f = MAX_f,\ n_f)$$

is added to the segment array to make $\Sigma(n/Flt)$ for that F2 segment an integer, where,

$$\begin{aligned} MIN_f = MAX_f &= (MIN_1 + MAX_1)/2 && \text{if} \quad (MIN_1 + MAX_1)/2 = (MIN_2 + MAX_2)/2 \\ &= MIN_1 && \text{if} \quad MIN_1 = MAX_2 \\ &= MAX_1 && \text{if} \quad MAX_1 = MAX_2 \\ &= (MIN_1 + MAX_1)/2 && \text{if} \quad \text{none above.} \end{aligned}$$

5. Next, selection must be made as to the type of flight sequence desired. Input IFRS and IAFS (card 2) and NFRS (card 12) if IFRS ≠ 0. The flight sequence may be:

   a) Random. IFRS=IAFS=0. The sequence is generated using a pseudo random number generator, see subroutine RANIC, page 88. The draw from the flight array is made without replacement. The standard seed number for the flight sequence random number generation is 11111 and is obtained by entering KF=0 in card 2. For a different random sequence enter 2051 < KF < 4194304 .

b) Alternate Sequence. Sequence according to the largest peak per flight. IFRS = 0

IAFS = 1 , LO-HI sequence
= 2 , HI-LO sequence
= 3 , LO-HI-LO sequence .

c) Specific sequence. IFRS = N , number of entries of 'flight type and number of flights' sets entered in card 12 (NFRS). Also, enter IAFS = 0 .

6. The spectrum loading sequence is generated by defining a sequence of valleys and peaks. Valey is MIN and peak is MAX as taken from the A6PA (MIN,MAX,n) array. A (valley, peak) pair is counted as a cycle, with valley preceding the peak. Groups of cycles are identified as a 'flight' according to NS (number of A6PA segments in each flight type) and F1 and F2 (sequence of these segments).

7. Choose one of two available valley/peak coupling methods:

a) IRS=1. The valley/peak pairs are defined directly from the program A6PA output (MIN_i, MAX_i, n_i) array pairs, where

$$\text{Valley} = MIN_i \quad \text{and} \quad \text{Peak} = MAX_i$$

The sequence of cycles (valley/peak pairs) so defined is random and is generated using a pseudo random number generator, see subroutine RANIC, page .

b) IRS=2. Valley = MIN and Peak = MAX are chosen individually and alternatingly in a random sequence from the program A6PA output (MIN,MAX,n) array. The random sequence is generated by a pseudo random number generator, see subroutine RANIC, page 88.

In both of the above cycle sequencing coupling methods in using the random number generator, the draw from the (MIN,MAX,n) array is made without replacement. The standard seed number for this application is 12345 and is obtained by entering KC=0 in card 2. For a different random sequence enter $2051 < KC < 4194304$.

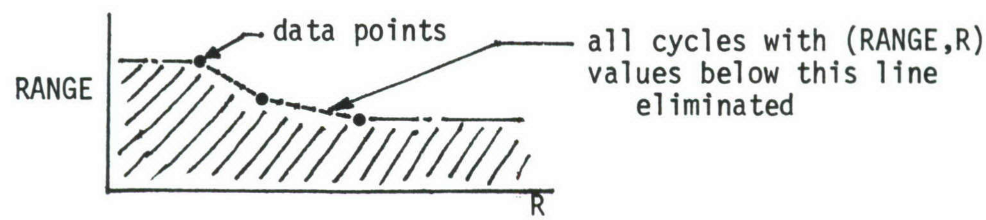
8. The cycle (valley, peak) sequence established in the previous steps is edited in the following order by EDIT1 and EDIT2 :

EDIT1

All valleys and peaks which are not loading reversal points are eliminated. This effectively eliminates fictitious load levels (if they do not produce a loading reversal) as well as intermediate data points.

EDIT2

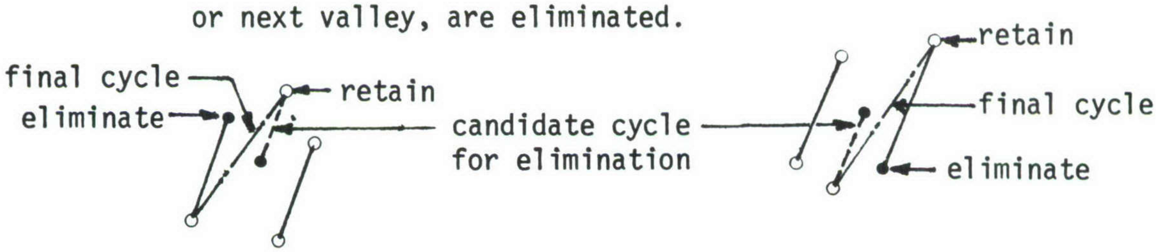
a) Range Truncation. Elimination of cycles as a function of cycle range (peak-valley) and R (valley/peak). If no range truncation is desired, NXY=0, card 2. If range truncation is to be performed, define the range truncation level over the desired R interval by XY input in card 9



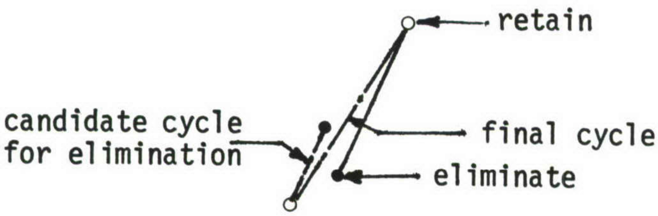
and specify the number of data points (RANGE,R sets) entered as NXY in card 2. The program interpolates linearly between the data points and will take the closest RANGE entry if cycle R is outside the R interval entered. In the latter case an ERROR statement will be printed.

There are two exceptions to this range truncation procedure:

1. When a peak or valley, from the cycle to be eliminated, is higher or lower than the preceding and next peak or valley. In such cases the subject peak or valley is retained, but their companion valley or peak, as well as the preceding peak or next valley, are eliminated.



2. When a cycle to be eliminated ($valley_i$, $peak_i$) is followed by a cycle with $valley_{i+1} > valley_i$ and $peak_{i+1} > peak_i$. In such case $peak_i$ and $valley_{i+1}$ are eliminated and the resulting cycle becomes $valley_i$, $peak_{i+1}$.



b) Cycle elimination if the peak is larger than a specified value ELIMP, card 3.

c) Clipping, according to specified peak (CLP) or valley (CLIV) magnitude, card 3. Given a cycle ($valley_i$, $peak_i$),

if $peak_i$ > CLP , set $peak_i$ = CLP
if $peak_i$ > CLP and $valley_i$ > CLP , eliminate cycle
if $valley_i$ < CLIV , set $valley_i$ = CLIV
if $valley_i$ < CLIV and $peak_i$ < CLIV, eliminate cycle.

d) Multiplication of the complete valley, peak sequence spectrum by a constant, FACTOR in card 3. Enter FACTOR=1 when multiplication is not desired. Multiplication will not be performed if an alternate (IAFS=1, 2 or 3) flight sequence is used.

9. If, through preceding editing, all cycles in a flight are eliminated, the flight will be counted in the total, but it will not be included in the output printout nor magnetic tape. An ERROR statement will be printed.

10. The basic output of program A6PD is a flight-by-flight loading sequence of valleys and peaks. For detail description of the output see the section on OUTPUT, page 101.

11. The program will terminate according to input IPTF (card 2): either when all flights (specified by NF, card 4) are processed or after N specified number of flights.

Variations in a spectrum can be accomplished through the various features provided in the A6PD input data as well as by changing the basic occurrences spectrum from A6PA through A6PA input data. Some of the variations that can be accomplished through the A6PD input data are:

1. Combining and/or elimination of A6PA segments.

2. Different Valley/Peak coupling.

3. Different flight sequences.

4 Elimination of cycles as a function of (RANGE, R), valley or peak values.

5. Clipping of loads below or above a specified value.

6. Multiplication of all loads by a constant.

7. Variation in spectrum length in terms of number of flights.

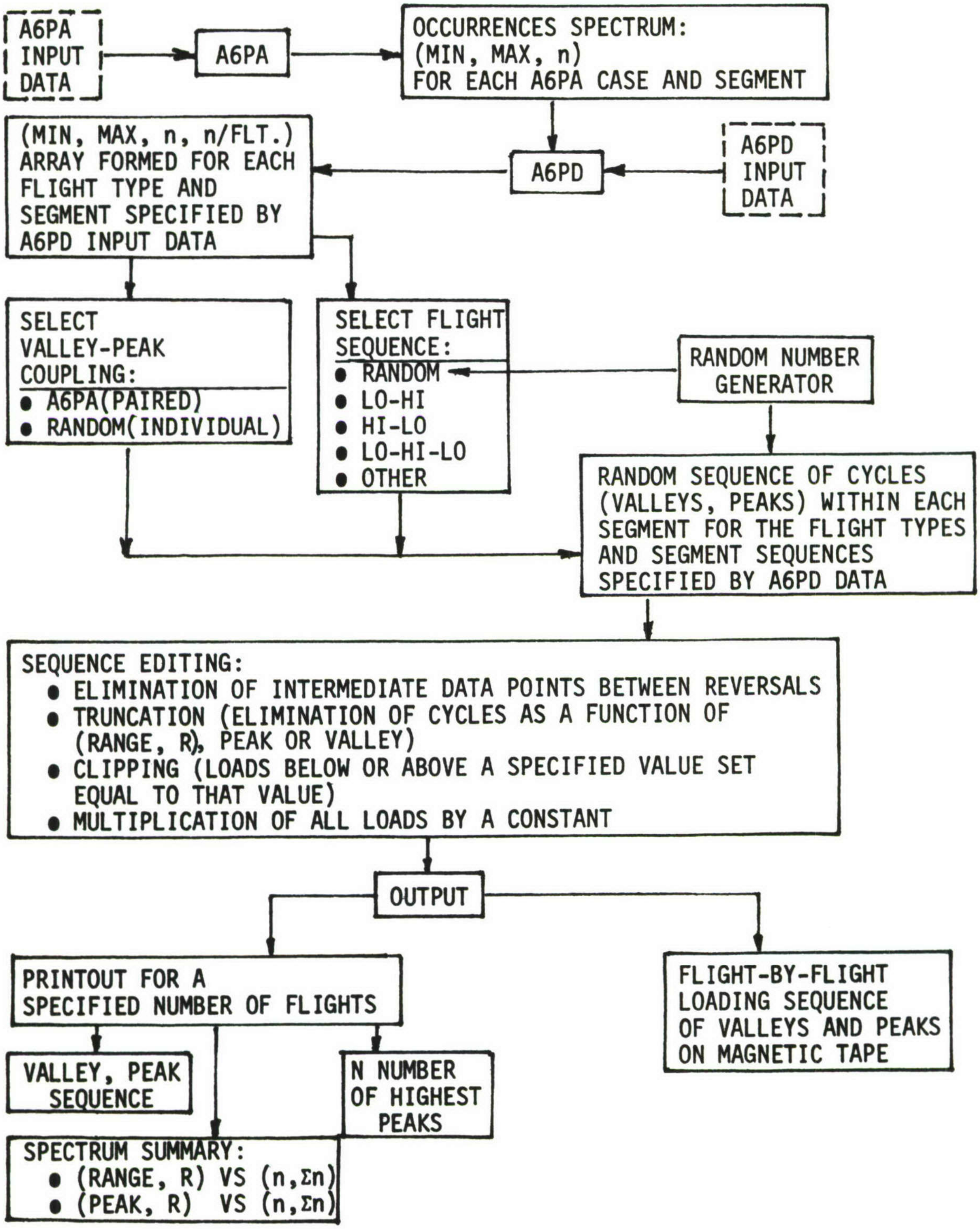


Figure 3 . Spectrum Loading Sequence Generation Program (A6PD) General Outline.

SECTION IV

DEFINITION OF SYMBOLS

The A6PD input data symbols and definitions are given in Section V. Following are the additional symbols used in the computer program. For program A6PA symbols and definitions see Reference 1 and Appendix A.

A (or N)	The dynamic working array. Throughout the program core allocation is done through implied equivalence to this array.
ACY	An array of cycles read in fron A6PA.
AFS	The reordered array of highest peak per flight (alternate flight sequence).
AMAX	An array of maximum stresses read in fron A6PA.
AMIN	An array of minimum stresses read in fron A6PA.
BCY	An array of cycles used during combining of segments.
HPEAK	An array of the highest peak per flight for printout.
ICASE	A6PA case number.
IDPEAK	An array of flight number identification for the highest peak per flight printout.
IERR	A running sum of the number of error messages written.
IFLAG	An array used in reordering the flight sequence to indicate when a flight has already been chosen.
IF1	An array of F1 segments.
IF2 (or IS2)	An array of F2 segments.
IJJ	An array used for debug that sums the number of cycles dropped out for the various editing cases.
IMM	A cycle counter used during edit type 1 and 2 .

IRR	A6PA reference run number, or, an array used for the cumulative totals in the spectrum summation table.
ISEG	A6PA segment number.
IS1	The beginning segment number for printout of flight type, reference run, case and segments, or, the beginning segment subscript of the IF1 and IF2 arrays.
IS2	The ending segment number for printout of flight type, reference, case and segments, or, the ending segment subscript of the IF1 and IF2 arrays.
JTN	An array of the flight types corresponding to a flight number.
KCY	An array of cycles after combining of segments, used when IRS = 2.
LEFT	The amount left of NSIZE usually calculated before a subroutine call after array sizes have been set.
LREC	An array of reordered flight numbers for the alternate flight sequence corresponding to the AFS array.
MCY	An array of the number of cycles per flight.
NCY	An array of cycles after combining of segments.
NFF	An array containing the sum of the number of times a flight type was picked and the sum of cycles for each flight type.
NPG	A page count. It is written at the top of each new output page.
NSIZE	The dimension of the "A" array.
NST	The total number of segments.
NTF	The total number of flights.

PMAX — An array of the highest peak per flight used for the alternate flight sequence.

PR — An array used for the spectrum summation print of peak vs. R .

RANGE — Cycle loading range = maximum stress (peak) - minimum stress (valley).

R — Stress ratio = minimum stress (valley) / maximum stress (peak).

RR — An array used for the spectrum summation print of range vs. R .

SMAX — An array of maximum stresses used during and after combining of segments.

SMIN — An array of minimum stresses used during and after combining of segments.

SMM — The final array of valley (minimum) and peak (maximum) stresses sequence.

SECTION V

INPUT DATA

The input data is divided into two groups:

1. Program A6PA data required to calculate the exceedances/occurrences spectra. This input data and data sheets are described in Reference 1 and Appendix A.

2. Sequence generation data (henceforth called "A6PD data") to generate the sequence of cycles and flights and to perform other manipulations in the A6PD program. The description of this input data and the sequence in which it is input follows.

CARD 1 - TITLE = Enter job description. Use only one card.

CARD 2 - NFT IRS IFRS IUIL NPI NRAN KVP IFI NXY IPFS NPSS IPTF IAFS MAXHP KF KC.

NFT	=	Number of flight types.
IRS	=	Valley/Peak coupling format.
	=	1 ; program A6PA pairs
	=	2 ; individual (random)
IFRS	=	Flight sequence.
	=	0 ; random. Also when IAFS $\neq$ 0.
	=	N ; number of 'flight type and number of flights' sets when special sequence is specified. Enter the 'flight type and number of flights' data on card(s) following F1 and F2 data (card 12).
IUIL	=	1 ; Fortran unit number
IPI NPI	=	Number* of peak levels in spectrum summation
IRAN NRAN	=	Number* of range levels in spectrum summation
IVP KVP	=	Number* of R levels in spectrum summation

* Enter the number as a negative number if the peak, range or R values are input directly in cards 6, 7 or 8. A positive number means that only the extreme values are entered in cards 6, 7 or 8 and the levels are calculated at equal intervals using the NPI, NRAN or KVP inputs.

IFI = Fortran unit number.
= 0 ; spectrum output tape not to be written.
= 3 ; spectrum output tape to be written.

NXY = Number of data points (R, range sets) in R vs range input data (card 9) if range truncation is to be performed. NXY=0 if no range truncation is to be performed (no data is to be input in card 9).

IPFS = Print loading sequence?
= -1 ; no.
= 0 ; yes, all flights.
= N ; yes, N flights.

NPSS = Number of spectrum summations to be printed. In card 10 indicate the flight number(s) for spectrum summation(s).

IPTF = Flight number at which the program is to be terminated.
= 0 ; normal, equals to total number of flights shown on card 4.
= N ; stop after N flights (less than the total shown on card 4).

IAFS = Alternate flight sequence.
= 0 ; random. Also, when IFRS=0 or N.
= 1 ; LO-HI (on the basis of the largest peak load per flight).
= 2 ; HI-LO
= 3 ; LO-HI-LO

MAXHP = Number of largest peaks per flight to be printed

IKF
KF = Seed number in the random number generator for sequencing flights.
= 0 ; will default to the number 11111 which is considered to be the standard number for this program.

IKC
KC = Seed number in the random number generator for sequencing cycles.
= 0 ; will default to the number 12345 which is considered to be the standard number for this program.

CARD 3 - CLIP CLIV FACTOR ELIMP

CLIP = Peak clipping value.

CLIV = Valley clipping value

FACTOR = Multiplication factor

ELIMP = Peak value for elimination of cycles with peaks above this value.

(Input very large numbers for CLIP, CLIV or ELIMP to prevent clipping or cycle elimination.)

CARD 4 - NF = Number of flights for each flight type that the A6PA data represents. Enter in increasing flight type order.

CARD 5 - NS = Number of A6PA segments in each flight type. Enter in increasing flight type order.

CARD 6 - PI = Peak values for spectrum summation. Enter all values if NPI in Card 1 is entered negative. Enter only the two extreme values if NPI in Card 1 is entered positive.

CARD 7 - RAN = Range values for spectrum summation. Enter all values if NRAN in Card 1 is entered negative. Enter only the two extreme values if NRAN in Card 1 is entered positive.

CARD 8 - VP = R values for spectrum summations. Enter all values if KVP in Card 1 is entered negative. Enter only the two extreme values if KVP in Card 1 is entered positive.

CARD 9 - XY = R vs range data (R, range sets equal to the number of sets specified by NXY in Card 1). Enter data in R increasing order.

CARD 10 - ISS = Flight number for spectrum summation. If making more than one entry, enter in increasing order.

CARD 11 - F1,F2 = Flags defining F1 and F2 segments. Enter F1 and then F2 flags for each flight type. Enter data in flight type increasing order. Start each set of F1 and F2 data for each flight on a new card.

CARD 12 - NFRS = Flight sequence when IFRS $\neq$ 0 in Card 2.
Enter flight type and number of flights sets in the desired sequence where the number of sets is equal to IFRS=N.

General Notes about A6PD Input Data:

1. The data consists of a string of values separated by one or more blanks or a comma. Blanks are not allowed within a constant. A decimal point omitted from a real constant is assumed to occur to the right of the rightmost digit of the mantissa.

2. Except for CARD 1 data, more than one card can be used to enter a particular data group identified as CARD X data in the preceding description. If more than one card is used for a given type of data, consider the cards as 6.1, 6.2, 6.3 or 11.1, 11.2, 11.3, etc. Input cards cannot have line sequence numbers.

3. Enter the data in the sequence indicated in the preceding writeup. This sequence is summarized on the following page in the form of an 'A6PD INPUT DATA FORMAT'.

An example of A6PD output input data printout for a sample case is shown on pages 208 and 209 .

A6PD INPUT DATA FORMAT

CARD	Columns 1–80
1	JØB TITLE
2	NFT IRS IFRS IUIL NPI NRAN KVP IFI NXY IPFS NPSS IPTF IAFS MAXHP KF KC
3	CLIP CLIV FACTØR ELIMP
4	NF = NUMBER ØF FLIGHTS FØR EACH FLIGHT TYPE
5	NS = NUMBER ØF A6PA SEGMENTS FØR EACH FLIGHT TYPE
6	PI = PEAK VALUES FØR SPECTRUM SUMATIØN
7	RAN = RANGE VALUES FØR SPECTRUM SUMMATIØN
8	VP = R VALUES FØR SPECTRUM SUMMATIØN
9	XY = R VS RANGE DATA
10	ISS = FLIGHT NUMBER FØR SPECTRUM SUMMATIØN
11.1	F1 DATA FØR FLIGHT TYPE 1
11.2	F2 DATA FØR FLIGHT TYPE 1
	(ETC., UNTIL F1 & F2 DATA IS ENTERED FØR ALL FLIGHT TYPES)
12	NFRS = FLIGHT SEQUENCE IF IFRS≠0

SECTION VI

PROGRAM LISTING

The following pages contain the program source language listing generated by the Fortran Extended (FTN) compiler.

Program A6PA (Ref. 1 and Appendix A) listing, as adapted to the CDC computer and A6PD program, is also presented here.

Figure 4 shows the interaction of all the subroutines. The listings of these subroutines, including additional descriptions, are to be found on the following pages:

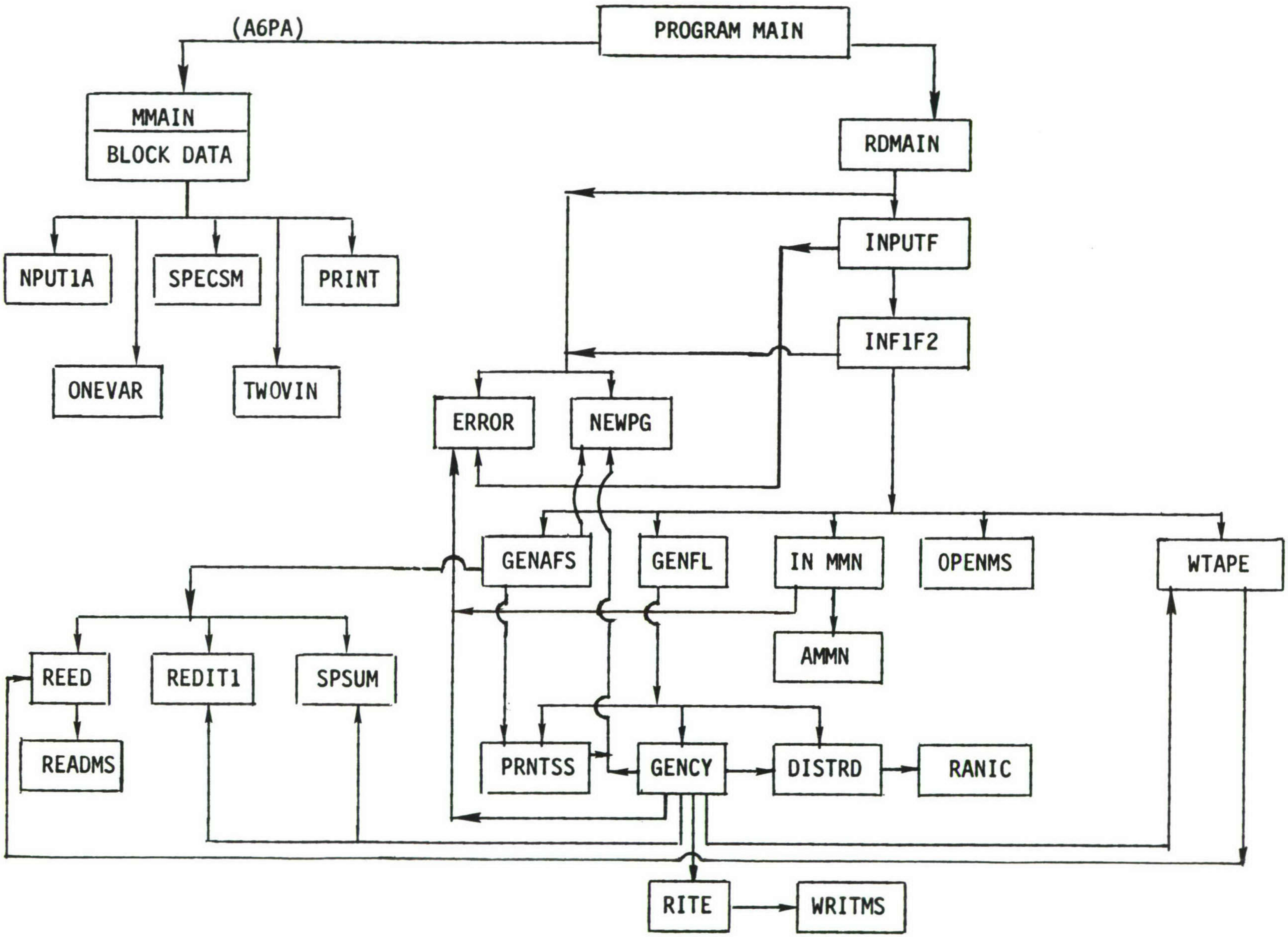


Figure 4. Program A6PD Subroutines.

```
      PROGRAM MAIN(INPUT,CUTPUT,TAPE1,TAPE3,TAPE4,TAPE5=INPUT,
     1            TAPE6=OUTPUT)
      REWINC 1
      CALL MMAIN
      WRITE(6,1)
    1 FORMAT (*1 CALLINC RANDCM PROGRAM*)
       REWIND 1
      CALL RDMAIN
      REWINC 1
      RETURN
      ENC
```

```
      SUBROUTINE MMAIN
      COMMON X, Y
      COMMON CSUM, CYCLSM, MAXN, CY, STSMXM, STSMNM
      COMMON /TABB/TKSIG
C***  DOUBLE PRECISION DATA
      DIMENSION YAW(40),CYBT(40),CYBTO(40)
      DIMENSION  X(3958), Y(3958), CDMGM(40), N(40), F(40), AM(40),
     1M5(40), DELTAY(25), DELT1(25), DELT2(25), DELT3(25), DELT4(25),
     2DELT5(25), DELT6(25), DELY1(40), DELY11(40), DY(40,25),
     3AMIDY(25),CDAMG(25), P(40), N1FLAG(40), YMAX(25), YMIN(25),
     4M3(40), CUMM(25), T(40), TAB1(25), TAB2(25), TAB3(25), TAB4(25),
     5TAB5(25), TAB6(25), ARNO1(40), SGMAX1(40), ARNO2(40), SGMAX2(40),
     6ARNO3(40), SGMAX3(40), ABR(40), VELOS(40), SLOPE(40), AKSIG(40),
     7WT(40), P1(40), AK1(40), P2(40), AK2(40), STSMXM(40,25),
     8STSMNM(40,25), CYCLSM(40,25), TABL1(25), TABL2(25), TABL3(25),
     9TABL4(25), TABL5(25), TABL6(25), TABL7(25), TABL8(25), TABL9(25)
      DIMENSION CSUM(40,25), K1(40), ISTRES(40), TBLM2(434), TBLI2(1542)
     1, IA(40), CYC(25), DMAGEM(25), ABC(40), N2(40), N6(40), QMAX(1000)
     2, QMIN(1000), JX(1000), JI(1000), MAXN(40), DATA(13), SCLTRB(40),
     3TKSIG(257), SIG(40)
      DIMENSION TBLSN(257), TBLLD(31)
      EQUIVALENCE (IEND, X(1)), (KEND, X(2)), (I4, X(3)),
     1(SIGULT, X(4)), (WAREA, X(5)), (ISTRES, X(6)), (DELT1, X(46)),
     2(TAB1, X(71)), (TAB2, X(96)), (TAB3, X(121)), (TAB4, X(146)),
     3(TAB5, X(171)), (AC, X(196)), (IW1, X(197)), (IW2, X(198)),
     4(IW3, X(199)), (IW4, X(200)), (IRR, X(201)), (ICASE, X(202)),
     5(IW5, X(203)), (TAB6, X(206)), (DELT2, X(231)), (DELY1, X(256)),
     6(DELY11, X(296)), (ARNO1, X(336)), (ARNO2, X(376)),
     7(ARNO3, X(416)), (SGMAX1, X(456)), (SGMAX2, X(496)),
     8(SGMAX3, X(536)), (T, X(576)), (AKSIG, X(616)), (SLOPE, X(656)),
     9(VELOS, X(696)), (WT, X(736)), (P1, X(776)), (TBLM2, X(853))
      EQUIVALENCE  (P2, X(1473)), (AK1, X(1513)), (AK2, X(1553)),
     1(ABR, X(1593)), (IA, X(1633)), (M3, X(1673)), (SIG, X(1713)),
     2(AM, X(1753)), (N, X(1793)), (NEND, X(1833)), (AL6, X(1850)),
     3(AL5, X(1851)), (AL4, X(1852)), (AL3, X(1853)), (AL2, X(1854)),
     4(AL1, X(1855)), (TBLI2, X(1856)), (M5, X(3398)), (N1FLAG, X(3438))
     5,(F,X(3478)), (N6, X(3518)), (N2, X(3558)), (P, X(3598)),
     6(SCLTRB, X(3638)), (TABL1, X(3678)), (TABL2, X(3703)),
     7(TABL3, X(3728)), (TABL4, X(3753)), (TABL5, X(3778)),
     8(TABL6, X(3803)), (TABL7, X(1287)), (TABL8, X(1312)),
     9(TABL9, X(1337)), (DELT3, X(1362)), (DELT4, X(1387))
      EQUIVALENCE (DELT5, X(1412)), (DELT6, X(1437)), (L1, X(1462))
      EQUIVALENCE   (CBART,X(3837)),(AST,X(3838)),(YAW,X(3839))
     L       ,(CYBT,X(3879)),(CYBTO,X(3919))
      REAL JX, JT, JSUM, ISUM, NEND, JI
   10 FORMAT(I3,I3)
   20 FORMAT(13A6)
   30 FORMAT(1H1)
   40 FORMAT(1H , 13A6)
      DO 50 K = 1,25
      DO 50 L = 1,40
```

```
      CSUM(L,K) = 0.0
      CYCLSM(L,K) = 0.0
   50 CONTINUE
      READ(5,10) IREAC, ICARD
      IF(IREAD.EQ.2) GO TO 85
      WRITE(6,30)
      LINENC = 0
      DO 80 K = 1, ICARC
      LINENC = LINENO + 1
      IF(LINENO - 50)60,6C,7C
   60 READ(5,20) (CATA(J) , J = 1,13)
      WRITE(6,40) (DATA(J) , J = 1, 13)
      GC TO 80
   70 WRITE(6,30)
      LINENC = 0
      GO TO 60
   80 CCNTINUE
   85 II = 0
   90 N3958 = 3958
      CALL NPUT1A (X(1), N3958, Y(1), II, IREF, ICAS, 0 )
  100 FCRMAT(1H1, 17HREFERENCE RUN NC. I6, 4X, 8HCASE NC. I6)
C********* WRITE REFERENCE RUN, CASE NC., AND SEGMENTS ON TAPE FOR *****
C********* SPECTRUM LOACING RANDOM SEQUENCE GENERATION PROGRAM      *****
      WRITE (3)  IRR,ICASE,IEND
      WLPRNT = 0.0
      IF(IW5 .EQ. 2) GO TO 11C
      CALL PRINT
      WLPRNT = 1.0
  110 B = 0.0
      K1(1) = 0
      TCDMGM = 0.0
  120 DO 1070 I = 1, IEND
      DC 130 KJ = 1, 257
  130 TBLSN(KJ) = 0.0
      DC 140 KJ = 1, 31
  140 TBLLD(KJ) = 0.0
      INTPER = 0
      IF(B .EQ. 1.0) GO TO 54C
      AX = 0.0
      CDMGM(I) = 0.0
      JEND = N(I)
      K = JEND - 1
      Q = 1.0
C     SELECTION IS MADE WHETHER OR NCT TO USE THE MULTIPLYING
C     FACTCR F(I)
      IF(L1 - 1)160,160,150
  150 D = 1.0
      GO TO 170
  160 D = F(I)
  170 IF((M3(I) .GT. 9) .AND. (M3(I) .LT. 13)) GC TC 380
C     LCAD SPECTRUM INPUT FORMAT IS SELECTED
      DC 260 J = 1, JENC
      M6 = M5(I)
```

```
      GC TO (180,190,20C,21C,220,230,240), M6
C     CALCULATE THE INCREMENTAL RESPCNSE DELTAY.
  180 DELTAY(J) = CELT1(J)
      GC TO 250
  190 DELTAY(J) = DELT2(J)
      GC TO 250
  200 DELTAY(J) = DELT3(J)
      GC TO 250
  210 DELTAY(J) = DELT4(J)
      GC TO 250
  220 DELTAY(J) = CELT5(J)
      GC TO 250
  230 DELTAY(J) = CELT6(J)
      GC TC 250
  240 DELTAY(J) = DELY1(I)    + DELY11(I)   *(Q - 1.0)
  250 Q = Q + 1.0
      DY(I,J) = DELTAY(J)
  260 CCNTINUE
      DC 310 J = 1, K
C     ESTABLISH MAX AND MIN RESPONSE VALUES AT MIDPCINTS BETWEEN
C     SUCCESSIVE DELTA Y VALLES
      AMIDY(J) = (DELTAY(J) + DELTAY(J + 1)) / 2.0
      IF(IW3 .EQ. 1) GO TC 27C
      P(I) = 0.0
  270 IF(N1FLAG(I) - 2)280,28C,29C
  280 YMAX(J) =D*(AM(I) + AMICY(J))+ P(I)
      IF(N1FLAG(I) .EQ. 2) GO TO 30C
      YMIN(J) = D*AM(I) + P(I)
      IF(N1FLAG(I) .EQ. 1) GC TO 31C
  290 YMAX(J) = D*AM(I) + P(I)
  300 YMIN(J) =D*(AM(I) - AMICY(J))+ P(I)
  310 CCNTINUE
      IF(M3(I) .LT. 13) GO TO 38C
      RHCO = 0.002378
      RHO1 = SIG(I) * RHOO
      IF(M3(I).EQ.14 .OR. M3(I).EC. 15 ) GO TO 380
      VAR  = 32.2 * AC * SLCPE(I) * RHO1
      WLCAD= 2.0 * WT(I) / WAREA
      FOUR = 4.0 * (WLOAD / VAR)
      PAR = FOUR + (6.28 / SLOPE(I))
      R1 = FOUR / PAR
      XARG = SCLTRB(I) / AC
      YARG = PAR
      XARGMN = TKSIG(18)
      YARGMN = TKSIG(2)
      IF(XARG - XARGMN)320,370,37C
  320 IF(YARG - YARGMN)330,35C,35C
  330 WRITE(6,340) XARG, YARG, NSEG
  340 FORMAT(1H0,45HGUST ALLEV. INTRP. ERROR. X IS TCC SMALL. X = E14.6,
     11X, 19HY IS TOO SMALL. Y = E14.6, 2X, 5HSEG =I2)
      GC TO 370
  350 WRITE(6,360) XARG, NSEG
  360 FCRMAT(1H0,45HGUST ALLEV. INTRP. ERROR. X IS TCC SMALL. X = E14.6,
```

```
     12X, 5HSEG = I2)
 370 NSEG = I + 50
     LEVEL = J
     CALL TWOVIN(XARG, YARG, TKSIG, OUTPUT, NSEG, LEVEL)
     AKSIG(I) = OUTPUT
     AKSIG(I) = R1 * AKSIG(I)
 380 DO 520 J = 1, JEND
C    CALCULATE THE CUMULATIVE CYCLES GIVEN VALUES OF DELTA Y.
     IF(M3(I) .EQ. 13) GO TO 460
     IF(M3(I).EQ.14.OR. M3(I).EQ.15 ) GO TO 455
     M1 = M3(I)
     GO TO(390,400,410,420,430,440,450,460,470,480,490,500), M1
 390 CUMM(J) = T(I) * TAB1(J)
     GO TO 510
 400 CUMM(J) = T(I) * TAB2(J)
     GO TO 510
 410 CUMM(J) = T(I) * TAB3(J)
     GO TO 510
 420 CUMM(J) = T(I) * TAB4(J)
     GO TO 510
 430 CUMM(J) = T(I) * TAB5(J)
     GO TO 510
 440 CUMM(J) = T(I) * TAB6(J)
     GO TO 510
 450 CUMM(J) = (ARNO1(I)*EXP(-DELTAY(J)**2/(2.0*(SGMAX1(I))   **2))
    1          + ARNO2(I)*EXP(-DELTAY(J)**2/(2.0*(SGMAX2(I)   )**2))
    2          + ARNO3(I)*EXP(-DELTAY(J)**2/(2.0*(SGMAX3(I)   )**2)))
    3          * T(I)
     GO TO 510
 460 ABR(I)  =  (VELOS(I)*SLOPE(I)*WAREA*AKSIG(I))/(498.0*WT(I))
     GO TO 470
 455 CONTINUE
 465 AMGT = (2*WT(I)/(RHO1*CBART*32.2*SLOPE(I)*AST)*(YAW(I)/WT(I)))/
    1       SCLTRB(I)**2
     AKSIG(I)=.88 * AMGT /(5.3 + AMGT )
     IF(M3(I).EQ.15) GO TO 466
     ABR(I) =  VELOS(I)*WAREA*AKSIG(I)/(498.*WT(I))*(CYBT(I)+CYBTO(I))
     GO TO 470
 466 ABR(I) =  (VELOS(I) * AST   *SLOPE(I)/ 498.) * AKSIG(I)
 470 CUMM(J) = (ARNO1(I)*P1(I)*EXP(-DELTAY(J)/(AK1(I)*ABR(I)   ))
    1          + ARNO2(I)*P2(I)*EXP(-DELTAY(J)/(AK2(I)*ABR(I)   )))
    2          * T(I)
     GO TO 510
 480 STSMXM(I,J) = TABL1(J)
     STSMNM(I,J) = TABL2(J)
     CYCLSM(I,J) = TABL3(J)
     K = JEND
     AX = 1.0
     GO TO 520
 490 STSMXM(I,J) = TABL4(J)
     STSMNM(I,J) = TABL5(J)
     CYCLSM(I,J) = TABL6(J)
     K = JEND
```

```
      AX = 1.0
      GC TO 520
  500 STSMXM(I,J) = TABL7(J)
      STSMNM(I,J) = TABL8(J)
      CYCLSM(I,J) = TABL9(J)
      K = JEND
      AX = 1.0
      GO TO 520
  510 CSUM(I,J) = CUMM(J)
  520 CCNTINUE
      K1(I) = K
      IF(AX .EQ. 1.0) GC TO 540
C     CALCULATE CYCLES FOR Y MAX AND Y MIN.
      DC 530 J = 1, K
      CYCLSM(I,J) = (CUMM(J) - CLMM(J + 1))
  530 CCNTINUE
  540 DC 940 J = 1, K
      IF(B .EQ. 1.0) GO TO 800
      IF(AX .EQ. 1.0) GC TO 72C
C     SELECT WHETHER TO ENTER CR NOT TO ENTER THE STRESS TABLES.
C     SELECTION IS MACE BY ISTRES FLAG.
      IF(ISTRES(I) .LT. 1) GO TC 61C
  550 NFLAG = 1
      ARGUMT = YMAX(J)
  560 M2 = ISTRES(I)
      M2 = (31 * M2) - 30
      DC 570 ITAB = 1, 31
      M20 = ITAB + M2 - 1
  570 TBLLD(ITAB) = TBLM2(M20)
C     SUBROUTINE ONEVAR - GIVEN A VALUE OF RESPONSE Y, INTERPOLATE IN
C     STRESS TABLES FOR A VALLE CF STRESS
      NSEGNM = I
      CALL CNEVAR(ARGUMT, TBLLD, CLTPLT, NSEGNM)
  580 GC TO(590,600), NFLAG
  590 STSMXM(I,J) = OUTPUT
      NFLAG = 2
      ARGUMT = YMIN(J)
      GC TO 560
  600 STSMNM(I,J) = OUTPUT
      GC TO 620
C     WHEN STRESS TABLES ARE NOT LSED, SET RESPONSE Y = STRESS
  610 STSMXM(I,J) = YMAX(J)
      STSMNM(I,J) = YMIN(J)
C     TEST TO ESTABLISH TRUE MAX AND MIN STRESS VALUES.
C     ALGEBRAICALLY, MAX STRESS GREATER THAN MIN STRESS.
  620 IF(STSMXM(I,J))64C,630,630
  630 IF(STSMNM(I,J))720,660,66C
  640 IF(STSMNM(I,J))65C,670,67C
  650 IF(ABS(STSMXM(I,J))- ABS(STSMNM(I,J)))690,690,670
  660 IF(STSMXM(I,J) - STSMNM(I,J))670,680,680
  670 SAVE = STSMXM(I,J)
      STSMXM(I,J) = STSMNM(I,J)
      STSMNM(I,J) = SAVE
```

```
680 IF(STSMXM(I,J) - 0.0)690,690,720
690 DMAGEM(J) = 0.0
    CYC(J) = 0.0
    CDAMG(J) = 0.0
    IF(B .EQ. 1.0) GO TO 1540
    IF(INTPER .EQ. 0) GO TO 940
    IF(YARG - YARGMN)700,940,940
700 NSEG = I
    LEVEL = J
    WRITE(6,710) XARG, YARG, NSEG, LEVEL
710 FORMAT(1H0, 32HSN INTP ERROR X IS TOO SMALL X = E14.6, 1X,
   118HY IS TOO SMALL Y = E14.6, 1X, 5HSEG = I2, 1X,
   212HLOAD LEVEL = I2, 1X, 16HDAMAGE SET = 0.0)
    INTPER = 0
    GO TO 940
C   FORM INTERPOLATING ARGUMENTS TO CALCULATE CYCLES TO FAILURE
C   FROM S-N DATA
720 IF(IA(I) - 6)740,740,730
730 IF((IA(I) .GT. 12) .AND. (IA(I) .LT. 19)) GO TO 750
740 XARG = (STSMXM(I,J) / SIGULT)
    IF(IA(I) - 6)760,760,770
750 XARG =((STSMXM(I,J) - STSMNM(I,J)) /(2.0 * SIGULT))
    IF(IA(I) - 18)770,770,780
760 YARG = (STSMNM(I,J) / STSMXM(I,J))
    GO TO 790
770 IF(IA(I) .GT. 18) GO TO 780
    YARG = ((STSMXM(I,J) + STSMNM(I,J)) / (2.0 * SIGULT))
    GO TO 790
780 YARG = (STSMNM(I,J) / SIGULT)
790 IF(IA(I) .LT. 7) I2 = IA(I)
    IF((IA(I) .GT. 6) .AND. (IA(I) .LT. 13)) I2 = (IA(I) - 6)
    IF((IA(I) .GT. 12) .AND. (IA(I) .LT. 19)) I2 = (IA(I) - 12)
    IF(IA(I) .GT. 18) I2 = (IA(I) - 18)
    ICALL = I2
800 I2 = (257 * I2) - 256
    DO 810  ISETTB = 1, 257
    I10 = ISETTB + I2 - 1
810 TBLSN(ISETTB) = TBLI2(I10)
    XARGMN = TBLSN(18)
    YARGMN = TBLSN(2) - 0.001
820 IF(XARG - XARGMN)830,840,840
830 INTPER = 1
    GO TO 690
840 NSEG = I
    LEVEL = J
    IF(B .EQ. 1.0) NSEG = 50
C   SUBROUTINE TWOVIN - LINEAR-QUADRATIC INTERPOLATION OF S-N DATA.
C   GIVEN THE INTERPOLATING VALUES XARG AND YARG, INTERPOLATE FOR A
C   VALUE OF CYCLES TO FAILURE.
    CALL TWOVIN(XARG, YARG, TBLSN, OUTPUT, NSEG, LEVEL)
850 GO TO(860,870,880,890,900,910), ICALL
860 ALIFE = AL1
    GO TO 920
```

```
  870 ALIFE = AL2
      GC TC 920
  880 ALIFE = AL3
      GC TC 920
  890 ALIFE = AL4
      GC TO 920
  900 ALIFE = AL5
      GO TC 920
  910 ALIFE = AL6
  920 IF(B .EQ. 1.0) GO TO 1530
      ALIFE = ALOG10(ALIFE)
      IF(OUTPUT - ALIFE)930,69C,69C
  930 CYC(J) = 10.0 ** CUTPUT
C     FORM THE RATIO DAMAGE = CYCLES EXPERIENCED AT A GIVEN RESPONSE
C     LEVEL / CYCLES TO FAILURE AT THAT RESPONSE LEVEL.
      DMAGEM(J) = CYCLSM(I,J) / CYC(J)
C     SUM THE DAMAGE DUE TO EACH LOAD INCREMENT WITHIN ONE SEGMENT.
      CDMGM(I) = CDMGM(I) + DMAGEM(J)
      CDAMG(J) = CDMGM(I)
  940 CCNTINUE
C     SUM THE DAMAGE OF ALL SEGMENTS.
      TCDMGM = TCDMGM + CDMGM(I)
C********* WRITE TAPE FCR                                        *********
C********* SPECTRUM LOADING RANDOM SEQUENCE GENERATION PROGRAM   *********
      JJEND = JEND - 1
      IF( (M3(I).GT.9).AND.(M3(I).LT.13) ) JJEND = JEND
      WRITE(1)JJEND,(STSMXM(I,J),STSMNM(I,J),CYCLSM(I,J),J=1,JJEND)
      IF(IW2 .EQ. 2) GO TO 1070
      WLPRNT = 1.0
      IF(IW5 .LT. 2) GO TO 950
      IF(I - 1)960,960,950
  950 WRITE(6,100) IRR, ICASE
  960 WRITE(6,970) I
  970 FORMAT(1H0, 4X, 9HSEGMENT = I2)
      WRITE(6,980)
  980 FORMAT(1H0, 51H-----------------------------------SPECTRUM---------
     127H--------------------------, 2X, 27H-----------DAMAGE CALCULATI
     213HON-----------)
      WRITE(6,990)
  990 FORMAT(1H , 5X, 1HJ, 3X, 7HDELTA Y, 3X,
     117HCUMULATIVE CYCLES, 2X, 10HMAX STRESS, 2X, 10HMIN STRESS,
     28X, 6HCYCLES, 10X, 9HALLOWABLE, 8X, 6HDAMAGE, 2X,
     310HCUM DAMAGE)
      DO 1050 J = 1, JEND
      IF((M3(I) .GT. 9) .AND. (M3(I) .LT. 13)) GC TC 1010
      WRITE(6,1000)     J, DELTAY(J), CUMM(J)
 1000 FORMAT(1H , 4X, I2,              F13.3, 1X, F16.4)
      IF(J .EQ. JEND) GC TO 1050
      IF((M3(I) .LT. 10) .OR. (M3(I) .EQ. 13)) GC TO 1030
      IF(M3(I).EQ.14 .OR. M3(I).EQ. 15 ) GO TC 1030
 1010 WRITE(6,1020) J, STSMXM(I,J), STSMNM(I,J), CYCLSM(I,J), CYC(J),
     1DMAGEM(J), CDAMG(J)
 1020 FORMAT(1H , 4X, I2, 32X, F10.0, 2X, F10.0, 2X, F16.4, 2X, F16.0,
```

```
     11X, F11.7, 1X, F11.7)
      GC TO 1050
 1030 WRITE(6,1040) STSMXM(I,J), STSMNM(I,J), CYCLSM(I,J), CYC(J),
     1MAGEM(J), CDAMG(J)
 1040 FCRMAT(1H , 38X, F10.0, 2X, F1C.C, 2X, F16.4, 2X, F16.0,
     11X, F11.7, 1X, F11.7)
 1050 CCNTINUE
      IF((M3(I).EQ.8).OR.(M3(I).EQ.9)) WRITE(6,1060) AKSIG(I), ABR(I)
      IF(M3(I).GE. 13 .AND. M3(I).LE. 15 ) WRITE(6,1060) AKSIG(I),ABR(I)
 1060 FORMAT(1H0, 5X, 25HGUST ALLEVIATICN FACTOR = F9.6, 5X,
     17HA-BAR = F16.6)
 1070 ABC(I) = TCDMCM
 1080 IF(I4.EQ.0) GC TO 1615
      CUMDMG = 0.0
      CUMXN  = 0.0
      JSUM = 0
      ISUM = 0
      I = 0
      J = 0
      M = 0
      K = 0
      K4= 0
      K5= 0
      DO 1160 I = 1, KEND
      N8 = K1(I)
      DO 1160 J = 1, N8
      IF(N6(I) - 1)1160,113C,1C9C
 1090 IF(K4 - 0)1110,1110,110C
 1100 IF(K5 - 0)1110,1110,112C
 1110 K = 0
 1120 K = K5
      K = K + 1
      QMAX(K) = STSMXM(I,J)
      JX(K)   = CYCLSM(I,J)
      K5 = K
      GC TO 1160
 1130 IF(K - K4)1140,1150,114C
 1140 K = K4
 1150 K = K + 1
      QMIN(K) = STSMNM(I,J)
      JI(K)   = CYCLSM(I,J)
      K4 = K
 1160 CCNTINUE
      IF(IW1 .EQ. 2) GO TO 12CC
      WLPRNT = 1.0
      WRITE(6,100) IRR, ICASE
      WRITE(6,1170)
 1170 FORMAT(1H0, 51HMAX AND MIN STRESSES AND CYCLE ARRAYS FCRMED FOR TH
     131HE DEFINITION OF THE GAG CYCLES.)
      WRITE(6,1180)
 1180 FCRMAT(1H , 51HARRAYS ARE FORMED FROM SEGMENTS AS SPECIFIED BY FLA
     17HG = N6.)
      WRITE(6,1190)
```

```
1190 FORMAT(1HO, 4X, 10HMAX STRESS, 10X, 6HCYCLES, 10X, 10HCUM CYCLES,
    110X, 10HMIN STRESS, 10X, 6HCYCLES, 10X, 10HCUM CYCLES)
1200 IF(K4 - K5)1210,1210,1220
1210 K = K5
     ILINE = 0
     GO TO 1230
1220 K = K4
     ILINE = 0
1230 DC 1390 I = 1, K
     IF(JSUM .GE. NEND) GO TO 1300
C    SCRT MAX ARRY INTO CESCENDING CRDER.
     DC 1240 J = I, K5
     IF(QMAX(I) .GE. QMAX(J)) GC TO 1240
     ST = CMAX(I)
     QMAX(I) = QMAX(J)
     QMAX(J) = ST
     JT = JX(I)
     JX(I) = JX(J)
     JX(J) = JT
1240 JCUT = I
     JSUM = JSUM + JX(I)
     IF(JSUM - NEND)1260,1250,1250
1250 JX(JOUT) = JX(JOUT) - (JSUM - NEND)
     JSUM = JSUM - (JSUM - NEND)
1260 IF(IW1 .EQ. 2) GO TO 1300
     WRITE(6,1270) QMAX(I), JX(I), JSUM
1270 FORMAT(1H , 4X, F10.0, 4X, F16.4, 3X, F16.4)
     IF(K4 - K5)1280,1280,1300
1280 ILINE = ILINE + 1
     IF(ILINE - 28)1300,1290,1290
1290 ILINE = 0
     WRITE(6,100) IRR, ICASE
     WRITE(6,1170)
     WRITE(6,1190)
1300 IF(ISUM .GE. NEND) GO TC 1390
C    SORT MIN ARRAY INTO ASCENCING CRDER.
     DC 1310 J = I, K4
     IF(QMIN(I) .LE. QMIN(J)) GO TO 1310
     ST = CMIN(I)
     QMIN(I) = QMIN(J)
     QMIN(J) = ST
     JT = JI(I)
     JI(I) = JI(J)
     JI(J) = JT
1310 ICUT = I
     ISUM = ISUM + JI(I)
     IF(ISUM - NEND)1330,1320,1320
1320 JI(IOUT) = JI(IOUT) - (ISUM - NEND)
     ISUM = ISUM - (ISUM - NEND)
1330 IF(IW1 .EC. 2) GO TO 1390
1340 WRITE(6,1350) QMIN(I), JI(I), ISUM
1350 FORMAT(1H , 60X, F10.0, 4X, F16.4, 3X, F16.4)
1360 IF(K4 - K5)1390,1390,1370
```

```
 1370 ILINE = ILINE + 1
      IF(ILINE - 28)1390,1380,1380
 1380 ILINE = 0
      WRITE(6,100) IRR, ICASE
      WRITE(6,1170)
      WRITE(6,1190)
 1390 CONTINUE
 1400 L = 1
      M = 1
C     FORM INTERPOLATING ARGUMENTS TO CALCULATE CYCLES TO FAILURE
C     FROM S-N DATA.
      IF(IW1 .EQ. 2) GO TO 1430
      ILINE = 0
      WRITE(6,100) IRR, ICASE
      WRITE(6,1410)
 1410 FORMAT(1H0, 41HGAG CYCLE SPECTRUM AND DAMAGE CALCULATION)
      WRITE(6,1420)
 1420 FORMAT(1H0, 5X, 10HMAX STRESS, 7X, 10HMIN STRESS, 9X, 6HCYCLES,
     19X, 10HCUM CYCLES, 9X, 1HR, 8X, 9HALLOWABLE, 7X, 6HDAMAGE,
     24X, 10HCUM DAMAGE)
 1430 IF(I4 - 6)1450,1450,1440
 1440 IF((I4 .GT. 12) .AND. (I4 .LT. 19)) GO TO 1460
 1450 XARG = (QMAX(L) / SIGULT)
      IF(I4 - 6)1470,1470,1480
 1460 XARG = (( QMAX(L) - QMIN(M)) /(2.0 * SIGULT))
      IF(I4 - 18)1480,1480,1490
 1470 YARG = (QMIN(M) / QMAX(L))
      GO TO 1500
 1480 IF(I4 .GT. 18) GO TO 1490
      YARG = ((QMAX(L) + QMIN(M)) / (2.0 * SIGULT))
      GO TO 1500
 1490 YARG = (QMIN(M) / SIGULT)
 1500 IF(I4 .LT. 7) I2 = I4
      IF((I4 .GT. 6) .AND. (I4 .LT. 13)) I2 = (I4 - 6)
      IF((I4 .GT. 12) .AND. (I4 .LT. 19)) I2 = (I4 - 12)
      IF(I4 .GT. 18) I2 = (I4 - 18)
      ICALL = I2
      IF(JX(L) .GE. JI(M))GO TO 1510
      JI(M) = JI(M) - JX(L)
      XN = JX(L)
      A = 1.0
      GO TO 1520
 1510 JX(L) = JX(L) - JI(M)
      XN = JI(M)
      A = 0.0
 1520 CUMXN = CUMXN + XN
      B = 1.0
      GO TO 120
 1530 ALIFE = ALOG10(ALIFE)
      IF(OUTPUT - ALIFE)1560,1540,1540
 1540 DMG = 0.0
      CYF = 0.0
      IF(YARG .GE. YARGMN) GO TO 1570
```

```
      WRITE(6,1550) XARG, YARG
 1550 FORMAT(1H0, 37HSN INTERP. ERROR. X IS TOO SMALL. X = E14.6 ,
     11X, 19HY IS TOO SMALL. Y = E14.6, 1X, 19HDAMAGE IS SET = 0.0, 1X,
     213H(GAG SEGMENT))
      GO TO 1570
 1560 CYF = 10.0 ** OUTPUT
      DMG = XN / CYF
 1570 CUMDMG = CUMDMG + DMG
      IF(IW1 .EQ. 2) GO TO 1600
      WRITE(6,1580)QMAX(L), QMIN(M), XN, CUMXN, YARG, CYF, DMG, CUMDMG
 1580 FORMAT(1H , 2X, F15.0, 2X, F15.0, 2X, F16.4, 2X, F16.4, 1X, F7.3,
     12X, F16.0, 1X, F11.7, 1X, F11.7)
      ILINE = ILINE + 1
      IF(ILINE - 54)1600,1590,1590
 1590 ILINE  = 0
      WRITE(6,100) IRR, ICASE
      WRITE(6,1410)
      WRITE(6,1420)
 1600 IF(L .EQ. JOUT .AND. M .EQ. IOUT) GO TO 1610
      IF(JX(L) .EQ. 0.0  .AND. A .EQ. 0.0) L = L + 1
      IF(A .EQ. 1.0) L = L + 1
      IF(A .EQ. 0.0) M = M + 1
      GO TO 1430
C     CALCULATION OF TOTAL DAMAGE INCLUDING GAG
 1610 TCDMGM = TCDMGM + CUMDMG
 1615 CONTINUE
      IF(IW4 .EQ. 2) GO TO 1620
      CALL SPECSM
      WLPRNT = 1.0
 1620 IF(WLPRNT .EQ. 0.0) GO TO 1650
      WRITE(6,100) IRR, ICASE
      WRITE(6,1630)
 1630 FORMAT(1H0, 43HINDIVIDUAL SEGMENT AND TOTAL DAMAGE SUMMARY)
      WRITE(6,1640)
 1640 FORMAT(1H0, 8X, 4HSEG., 7X, 6HDAMAGE, 11X, 5HTOTAL)
      GO TO 1670
 1650 WRITE(6,1630)
      WRITE(6,1660)
 1660 FORMAT(1H , 8X, 4HSEG., 7X, 6HDAMAGE, 11X, 5HTOTAL)
 1670 WRITE(6,1680) (I, CDMGM(I), ABC(I), I = 1, IEND)
 1680 FORMAT(1H , 9X, I2, F16.7, F16.7)
      IF(I4 .EQ. 0) GO TO 1700
      WRITE(6,1690) CUMDMG, TCDMGM
 1690 FORMAT(9X, 3HGAG, F16.7, F16.7)
 1700 IF(II.EQ.1) GO TO 90
      RETURN
      END
      BLOCK DATA
      COMMON /TABB/TKSIG
      DIMENSION TKSIG(257)
      DATA TKSIG          /15.0,10.0,20.0,30.0,40.0,50.0,70.0,100.0,
     1150.0,300.0,500.0,1000.0,0.0,0.0,0.0,0.0,15.0,6.0,10.0,14.0,18.0,
     222.0,30.0,40.0,60.0,80.0,100.0,120.0,140.0,160.0,180.0,240.,2.954,
```

```
32.69,2.515,2.295,2.162,1.972,1.82,1.622,1.514,1.413,1.349,1.289,  00005820
41.259,1.231,1.162,3.8,3.59,3.31,3.02,2.82,2.514,2.19,1.884,1.719, 00005830
51.597,1.48,1.413,1.35,1.303,1.202,4.22,4.075,3.8,3.51,3.315,2.95, 00005840
62.632,2.19,1.998,1.82,1.68,1.604,1.513,1.446,1.319,4.46,4.36,4.14,00005850
73.8,3.635,3.275,2.884,2.51,2.29,2.02,1.862,1.74,1.7,1.64,1.48,     00005860
84.68,4.63,4.36,4.15,3.98,3.55,3.16,2.755,2.483,2.24,2.09,1.95,     00005870
91.862,1.78,1.62,5.012,5.065,4.9,4.67,4.46,4.07,3.63,3.16,2.85,     00005880
X2.63,2.4,2.24,2.09,1.995,1.8,5.346,5.52,5.5,5.37,5.14,4.67,4.26,   00005890
A3.72,3.35,3.02,2.755,2.632,2.514,2.345,2.09,5.624,6.026,6.05,6.03,00005900
B5.89,5.63,5.14,4.52,4.07,3.645,3.39,3.125,2.92,2.758,2.458,6.026, 00005910
C6.457,6.746,6.903,6.919,6.839,6.607,5.95,5.37,4.9,4.56,4.26,4.07, 00005920
D3.8,3.315,6.096,6.684,6.919,7.228,7.328,7.345,7.245,6.887,6.457,  00005930
E5.95,5.63,5.25,5.01,4.74,4.16,6.166,6.839,7.145,7.413,7.586,7.727,00005940
F7.763,7.586,7.379,7.079,6.808,6.562,6.309,6.03,5.31, 60*0.0/      00005950
  END                                                              00005960
```

```
      SUBROUTINE NPUT1A ( CASE, NCASE, RRAREA, IENTRY, IREF, ICAS, KP ) 00000015
CDECK NPTA           REVISED AND CHECKED BY D B KNUDSEN      05/14/70   00000020
                                 CHECKED BY D B KNUDSEN      25 FEB 76   00000030
                        TO ALLOW USE OF FORTRAN H-EXTENDED                 00000040
C                                                                        00000050
C     NPUT1A - STANDARD DATA INPUT                                       00000060
C                                                                        00000070
C     THIS SUBROUTINE READS A STANDARD DATA FORM (VARIABLE NUMBER OF     00000080
C        CARDS WITH UPTO FOUR VALUES PER CARD) AND STORES VALUES         00000090
C        (INTEGER AND/OR REAL) INTO AN ARBITRARY LENGTH ARRAY.  A        00000100
C        PROCEDURE FOR EITHER REPLACING OR UPDATING REFERENCE RUN        00000110
C        ARRAYS IS ALSO PROVIDED.                                        00000120
C                                                                        00000130
C                                                                        00000150
C        CASE   = OUTPUT = A LINEAR ARRAY (SEE Y) CONTAINING THE DATA    00000160
C                          READ FROM CARDS                               00000170
C                                                                        00000180
C        NCASE  = IN-OUT = THE UPPER LIMIT OF THE CASE ARRAY             00000190
C                                                                        00000230
C        RRAREA = OUT-IN = A REFERENCE RUN ARRAY (OF THE SAME LENGTH     00000240
C                          AS CASE) IF  RRAREA IS NOT EQUIVALENT TO      00000250
C                          CASE.  THE SUBROUTINE SAVES DATA IN THIS      00000260
C                          AREA BETWEEN CALLS.  IF  RRAREA  IS           00000270
C                          EQUIVALENCED TO  CASE,  THEN RRAREA IS NOT    00000280
C                          USED AND THERE CANNOT BE ANY REFERENCE RUN    00000290
C                          DATA                                          00000300
C                                                                        00000310
C        IENTRY = IN-OUT = ENTRY FLAG (INTEGER)                          00000320
C                        =  0  INITIAL ENTRY ONLY (INPUT)                00000330
C                        =  2  NEW ENTRY ONLY (INPUT) TO CHANGE CASE,    00000340
C                                 Y, AND/OR RRAREA.  NO INFORMATION      00000350
C                                 IS SAVED FROM PREVIOUS CASE OR         00000360
C                                 RRAREA ARRAYS.                         00000370
C                        =  3  OVERLAY RRAREA (INPUT).  PERMITS          00000380
C                                 OVERLAYING OF REFERENCE RUN DATA.      00000390
C                        =  1  NORMAL OUTPUT FLAG (OUTPUT)               00000400
C                        = -1  LAST CASE FLAG (OUTPUT).  THE NEXT        00000410
C                                 ENTRY TO THE SUBROUTINE WILL           00000420
C                                 TERMINATE YOUR JOB.                    00000430
C                                                                        00000440
C        IREF   = OUTPUT = REFERENCE RUN NUMBER (INTEGER) TAKEN FROM     00000450
C                          INPUT CARDS                                   00000460
C                                                                        00000470
C        ICAS   = OUTPUT = CASE NUMBER (INTEGER) TAKEN FROM DATA         00000480
C                          CARDS                                         00000490
C                                                                        00000500
C        KP     = INPUT = OPTIONAL PRINT CODE (INTEGER)                  00000510
C                         IF KP IS NOT IN ARGUMENT LIST, IT IS           00000520
C                         TREATED AS IF KP = 0                           00000530
C                       = 0   EJECT PAGE FOR EACH NEW DATA CASE          00000540
C                       = 1   SINGLE SPACE FOR EACH NE DATA CASE         00000550
```

```
C                         = ANYTHING ELSE - NO PRINTING AT ALL
C
C     THIS ROUTINE USES Z7L1 AND ARGQ
C                       THE IABS AND ISIGN FUNCTIONS
C
C     FORTRAN UNIT 5 IS READ (FOR DATA)
C     FORTRAN UNIT 6 MAY BE WRITTEN (SEE INPUT PARAMETER KP) WITH
C         REFERENCE RUN AND CASE NUMBERS AT THE TOP OF A NEW PAGE (OR
C         AFTER ONE BLANK LINE) AFTER EACH EXECUTED CALL STATEMENT.
C         ERROR MESSAGES MAY ALSO BE WRITTEN.
C
C     CARD FORM IS
C            CC   1      MUST CONTAIN A 1
C            CC   2- 6,  18-22, 34-38, 50-54   LOCATION FIELDS
C            CC   7-15,  23-31, 39-47, 55-63   FRACTION FIELDS
C            CC   16-17, 32-33, 48-49, 64-65    EXPONENT FIELDS
C            CC   66-68   UNUSED
C            CC   69-70   REFERENCE RUN NUMBER
C            CC   71-73   CASE NUMBER
C            CC   74-80   UNUSED
C
      INTEGER  CASENO, CASNUM, ENTRY1, REFNO, REFNUM
C
      LOGICAL  NER1, NER2, NER3, REFRUN
C
      DOUBLE PRECISION A
C
      DIMENSION  CASE(1), RRAREA(1)
      DIMENSION  ITB(21), ITN(20), IE1(4), IE2(4), IS(4), IV(4), L(4)
C
      EQUIVALENCE (AN,N)
C
      DATA  ITB / 1H0, 1H1, 1H2, 1H3, 1H4, 1H5, 1H6, 1H7, 1H8, 1H9,
     1            1H , 1HJ, 1HK, 1HL, 1HM, 1HN, 1HO, 1HP, 1HQ, 1HR, 1H /
      DATA  ITN /  0,  1,  2,  3,  4,  5,  6,  7,  8,  9,
     1            -0, -1, -2, -3, -4, -5, -6, -7, -8, -9 /
      DATA  IX  / 1HX /
C
   10 FORMAT (I1, 4(I5, A1, I8, 2A1), 3X, I2, I3)
   20 FORMAT (3H0ER I2, 5X I1, 4(I5,A1,I8,2A1), 3X I2, I3)
   30 FORMAT ( A1, 17HREFERENCE RUN NO.I3, 4X, 8HCASE NO.I4 )
   40 FORMAT (48H1NO MORE INPUT1 DATA...JOB TERMINATED BY INPUT1.)
C
      ENTRY1 = IENTRY
      NER1 = .FALSE.
      NER2 = .FALSE.
      NER3 = .FALSE.
C
C     DETERMINE NUMBER OF ARGUMENTS IN CALL STATEMENT
C
C***  CALL ARGQ ( N )
C
C     SET PRINT FOR PAGE EJECT, CHECK NUMBER OF ARGUMENTS
```

```
C
      KKKK = 2
C***  IF (N .LT. 7) GO TO 5C
C
C     KP INCLUDED. CHECK FOR PAGE EJECT
C
      IF  ( KP .EQ. 0 )    GO TO 5C
C
C     PAGE EJECT NOT WANTED. SET PRINT FOR SINGLE SPACE, CHECK KP
C
      KKKK = 21
      IF (KP .NE. 1) KKKK = 50
C
C     TEST ENTRY FLAG
C
   50 IF  ( ENTRY1 .EQ. -1 )   GO TO 470
      NER = 0
      IF  ( ENTRY1 .EQ. 2 )  GO TO 60
      IF (ENTRY1 .NE. 0) GO TO 2CC
   60 REFRUN = .FALSE.
C
C     CHECK FOR NO REFERENCE RUN ARRAY
C
C***  CALL ZZL1 ( CASE, RRAREA, LADC, LBDD, N )
C***  IF  ( LADC .NE. LBCC )  REFRUN = .TRUE.
C***  CALL ZZL1 ( CASE, Y, LADC, LBDD, N )
C***  NLL = N
      IF( NCASE.GT.0 )  REFRUN = .TRUE.
      NLL = IABS(NCASE)
      IF  ( ENTRY1 .EQ. 2)  GO TO 190
C
C     READ INPUT DATA CARD, TEST FOR END CARD AND ENTRY FLAG, CHECK
C          REFERENCE RUN AND CASE NUMBERS (FOR ALL BUT FIRST ENTRY)
C
   70 READ (5,10) IC1, ( L(I), IS(I), IV(I), IE1(I), IE2(I), I = 1, 4 ),
     1            REFNO, CASENO
      IF  ( REFNO .EQ. 99  .AND.  CASENO .EQ. 999 )  NER3 = .TRUE.
      IF  ( ENTRY1 .EQ. 0 )  GO TO 190
      IF  ( CASNUM .NE. 0 )  GO TO 8C
      IF  ( CASENO .NE. 0 )  GO TO 140
      IF  ( REFNUM .NE. REFNO )  GO TO 460
      GO TO 110
C
C     TEST FOR END OF CURRENT CASE
C
   80 IF  ( CASNUM .EQ. CASENO )  GO TO 170
      IF  ( NER3 )  ENTRY1 = -1
C
C     CHECK FOR PAST ERRORS
C
      IF  ( NER1 .OR. NER2 )  GO TO 100
C
C     PRINT (IF KP = 0 OR 1 ) OUTPUT TITLE AND EXIT FROM SUBROUTINE
```

```
C
      IF  ( KKKK .GT. 40 )  GO TO 9C
      WRITE  (6,30)  ITB(KKKK), REFNUM, CASNUM
   90 IREF = REFNUM
      ICAS = CASNUM
      IENTRY = ENTRY1
      RETURN
C
C     RESET ERRCR FLAG1 AND TEST FOR RESETTING ERRCR FLAG2
C
  100 IF  ( NER2  .AND.  (REFNC - REFNUM) .NE. 0 )  NER2 = .FALSE.
      NER1 = .FALSE.
      GC TO 50
C
C     TEST FOR REFERENCE RUN DATA AND ARRAY
C
  110 IF  ( REFNO .EQ. C )  GC TC 120
      IF  ( .NOT. REFRUN )  GO TO 440
      GC TO 150
C
C     SET CASE ARRAY TO ZERC (NO REFERENCE RUN DATA)
C
  120 DO  130  I = 1, NLL
  130 CASE(I) = 0.0
      REFNUM = 0
      GC TO 180
C
C     MCVE REFERENCE RUN ARRAY INTO CASE ARRAY
C
  140 CASNUM = CASENO
  150 DO  160  I = 1, NLL
  160 CASE(I) = RRAREA(I)
C
  170 IF  ( REFNUM .NE. REFNO )  GO TO 460
  180 CASNUM = CASENO
      GC TO 220
C
C     INITIAL ENTRY ( ENTRY1 = 0 CR 2)
C
  190 ENTRY1 = 1
      REFNUM = -1
      CASNUM = -1
C
C     TEST REFERENCE RUN AND CASE NUMBERS (NORMAL REENTRY)
C
  200 IF  ( CASENO .NE. 0 )  GC TC 110
      IF  ( REFNO .EQ. 0 )  GO TO 450
      IF  ( .NOT. REFRUN )  GC TC 440
C
C     SET UP REFERENCE RUN AND CASE NUMBER CF NEW REFERENCE RUN
C
      REFNUM = REFNO
      CASNUM = 0
```

```
C
C     CHECK FOR OVERLAY FLAG (AND IF NOT, SET REFERENCE RUN ARRAY = 0 )
C
      IF  ( ENTRY1 .EQ. 3 )  GO TO 220
      DO  210  I = 1, NLL
  210 RRAREA(I) = 0.0
C
C     RESET ENTRY FLAG AND TEST COLUMN ONE OF DATA CARD
C
  220 ENTRY1 = 1
      IF  ( IC1 .NE. 1 )  GO TO 430
C
C     CONVERT, CHECK, AND (IF CORRECT) STORE FOUR SETS OF DATA FIELDS
C
      DO  350  I = 1, 4
      J = L(I)
      JS = IS(I)
      JV = IV(I)
C
C     CHECK SIGN FIELD (AND IF BLANK, SKIP TO NEXT SET OF FIELDS)
C
      IF  ( JS .EQ. ITB(21) )  GO TO 350
C
C     TEST LOCATION FOR VALID RANGE
C
      IF  ( J .LE. 0 )  GO TO 370
      IF  ( J .GT. NLL )  GO TO 360
C
C     FIND AND CHECK SIGN FIELD VALUE
C
      K = 1
  230 IF  ( JS .EQ. ITB(K) )  GO TO 240
      K = K + 1
      IF  ( K .NE. 21 )  GO TO 230
      GO TO 400
C
C     CHECK FOR VALID VALUE FIELD - CONVERT AND STORE SIGN AND VALUE
C
  240 IF  ( JV .LT. 0 )  GO TO 390
      IF  ( K .NE. 11 )  GO TO 250
      N = -JV
      GO TO 260
  250 N = ITN(K)
      N = ISIGN( 100000000 * IABS(N) + JV, N )
C
C     CHECK EXPONENT FOR FLOATING POINT
C
  260 IF  ( IE1(I) .EQ.IX  .AND.  IE2(I) .EQ.IX )  GO TO 330
C
C     CONVERT AND CHECK FLOATING EXPONENT
C
      K = 1
  270 IF  ( IE2(I) .EQ. ITB(K) )  GO TO 280
```

```
      K = K + 1
      IF  ( K .EQ. 21 )  GO TO 420
      GO TO 270
  280 N2 = ITN(K)
      IF  ( K .GT. 10 )  GO TO 410
C
      K = 1
  290 IF  ( IE1(I) .EQ. ITB(K) )  GO TO 300
      K = K + 1
      IF  ( K .EQ. 21 )  GO TO 420
      GO TO 290
  300 IF ( K .NE. 11 )  GO TO 310
      N3 = -N2
      GO TO 320
  310 N1 = ITN(K)
      N3 = ISIGN( 10 * IABS( N1 ) + N2, N1 )
C
  320 IF  ( N3 .LT. (-60)  .OR.  N3 .GT. 70 )  GO TO 380
C
C     CONVERT VALUE (N) TO FLOATING POINT (USING EXPONENT)
C
      A = N
      AN = A * ( 10.0 ** (N3 - 9) )
C
C     STORE ANSWER IN LOCATION J OF REFERENCE RUN OR CASE ARRAYS
C
  330 IF  ( CASENO .EQ. 0 )  GO TO 340
      CASE(J) = AN
      GO TO 350
  340 RRAREA(J) = AN
  350 CONTINUE
      GO TO 70
C
C     SET ERROR CODE
C
  360 NER = NER + 2
  370 NER = NER + 2
  380 NER = NER + 2
  390 NER = NER + 1
  400 NER = NER + 1
  410 NER = NER + 1
  420 NER = NER + 1
  430 NER = NER + 2
  440 NER = NER + 1
  450 NER = NER + 1
  460 NER = NER + 1
      NER1 = .TRUE.
      IF (REFRUN .AND. CASENO .EQ. 0 ) NER2 =.TRUE.
C
C     WRITE ERROR MESSAGES
C
      IF  ( NER3 )  GO TO 470
      WRITE  (6,20)  NER, IC1, ( L(I), IS(I), IV(I), IE1(I), IE2(I),
```

```
     1                 I = 1, 4 ), REFNC, CASENC
      NER = 0
      GC TO 70
C
C     TERMINATE WHEN ALL CATA HAS BEEN READ
C
  470 WRITE (6,40)
      STCP
      END
```

```
      SUBROUTINE SPECSM
      COMMON X, Y
      COMMON CSUM, CYCLSM, MAXN, DY, STSMXM, STSMNM
      DIMENSION X(3958), Y(3958)
      DIMENSION CSUM(40,25), CYCLSM(40,25), N2(40), MAXN(40), M3(40),
     1DY(40,25), N(40), STSMXM(40,25), STSMNM(40,25)
      EQUIVALENCE  (IEND, X(1)),  (IRR, X(201)), (ICASE, X(202))
      EQUIVALENCE (M3, X(1673)), (N, X(1793)), (N2, X(3558))
   10 FORMAT(1H1, 17HREFERENCE RUN NO. I6, 4X, 8HCASE NO. I6)
      I = 1
      L = 0
      L5= 0
      M = 1
      M9= 0
      M10 = 0
   20 K = 0
      IF(N2(I) .EQ. 0) GO TO 130
      L = L + 1
      L5= L5 + 1
      N(L) = N(I)
      MAXN(L) = N(I)
      M3(L) = M3(I)
      JEND = N(L)
      DO 30 J = 1, JEND
      K = K + 1
      DY(L,K) = DY(I,J)
      CSUM(L,K) = CSUM(I,J)
      IF((M3(I) .GT. 9).AND.(M3(I).LT.13))CSUM(L,K) = CYCLSM(I,J)
   30 CONTINUE
   40 M = M + 1
      M10 = 0
      IF(N2(I) .EQ. 0) GO TO 130
      IF(M - IEND)50,50,130
   50 IF(N2(I) - N2(M))40,60,40
   60 IF(I - M)70,20,130
   70 K = 0
      IF(N(L) - N(M))80,110,100
   80 N(L) = N(M)
      M9 = 1
      IF(MAXN(L) - N(M))90,110,110
   90 MAXN(L) = N(M)
      M10 = 1
      GO TO 110
  100 N(L) = N(M)
      M9 = 2
  110 JEND = N(L)
      DO 120 J = 1, JEND
      K = K + 1
      IF((M9 .EQ. 1) .AND. (M10 .EQ. 1)) DY(L,K) = DY(M,J)
      CSUM(L,K) = CSUM(L,K) + CSUM(M,J)
      IF((M3(I).GT.9).AND.(M3(I).LT.13))CSUM(L,K)=CSUM(L,K)+CYCLSM(M,J)
```

```
120 CONTINUE
    GC TO 40
130 I = I + 1
    M = 0
    IF(I .LE. IEND) GC TO 40
    DC 260 L = 1, L5
    IF((M3(L) .LT. 10) .OR. (M3(L) .GT. 12)) GO TO 160
    WRITE(6,10) IRR, ICASE
    WRITE(6,140)
140 FORMAT(1H0, 50HTHE FOLLOWING DATA IS THE SUMMATICN OF THE SPECTRA
   138H FOR SEGMENTS SPECIFIEC BY FLAG L = N2)
    WRITE(6,150)
150 FORMAT(1H0, 3X, 1HL, 3X, 2HLL,5X, 10HMAX STRESS, 5X,
   110HMIN STRESS, 12X, 6HCYCLES)
    GO TO 190
160 WRITE(6,10) IRR, ICASE
    WRITE(6,140)
170 WRITE(6,180)
180 FORMAT(1H0, 3X, 1HL, 3X, 2HLL, 5X, 7HDELTA Y, 5X,
   117HCUMULATIVE CYCLES, 10X, 6HCYCLES)
190 IF(N(L) .LT. MAXN(L)) N(L) = MAXN(L)
    JEND = N(L)
    KJ = N(L) - 1
    DO 200 K = 1, KJ
200 CYCLSM(L,K) = CSUM(L,K) - CSUM(L, K+1)
    DO 260 K = 1, JEND
    IF((M3(L) .GT.  9) .AND. (M3(L) .LT. 13)) GO TO 220
    WRITE(6,210) L, K, DY(L,K), CSUM(L,K)
210 FORMAT(1H , 2X, I2, 3X, I2, 2X, F13.3, 3X, F16.4)
    IF(K .EQ. JEND) GC TO 26C
    IF((M3(L) .LT. 10) .OR. (M3(L) .GE. 13)) GC TO 240
220 WRITE(6,230) L, K, STSMXM(L,K), STSMNM(L,K), CSUM(L,K)
230 FORMAT(1H , 2X, I2, 3X, I2, 4X, F11.0, 4X, F11.0, 7X, F16.4)
    GC TO 260
240 WRITE(6,250) CYCLSM(L,K)
250 FORMAT(1H , 46X, F16.4)
260 CONTINUE
    DO 270 L = 1, 40
    DC 270 K = 1, 25
270 CSUM(L,K) = 0.0
    RETURN
    END
```

```
      SUBROUTINE PRINT
      COMMON X, Y
      DIMENSION X(3958), Y(3958), SIG(40), SCLTRB(40), F(40)
      DIMENSION    YAW(40),CYBT(40),CYBTO(40)
      DIMENSION M3(40), M5(40), ISTRES(40), IA(40), N1FLAG(40), P(40),
     1N6(40), N2(40), AM(40), DELY1(40), DELY11(40), ARNO1(40),
     2ARNO2(40), ARNO3(40), SGMAX1(40), SGMAX2(40), SGMAX3(40),
     3AKSIG(40), SLOPE(40), VELOS(40), WT(40), P1(40), P2(40), AK1(40),
     4AK2(40),            ABR(40), N(40), TBLM2(434), TBLI2(1542)
      DIMENSION DELT1(25), DELT2(25), DELT3(25), DELT4(25), DELT5(25),
     1DELT6(25), TAB1(25), TAB2(25), TAB3(25), TAB4(25), TAB5(25),
     2TAB6(25), TABL1(25), TABL2(25), TABL3(25), TABL4(25), TABL5(25),
     3TABL6(25), TABL7(25), TABL8(25), TABL9(25), T(40)
      EQUIVALENCE (IEND, X(1)), (KEND, X(2)), (I4, X(3)), (SIGULT, X(4))
     1, (WAREA, X(5)), (ISTRES, X(6)), (DELT1, X(46)), (TAB1, X(71)),
     2(TAB2, X(96)), (TAB3, X(121)), (TAB4, X(146)),
     3(TAB5, X(171)), (AC, X(196)), (IW1, X(197)), (IW2, X(198)),
     4(IW3, X(199)), (IW4, X(200)), (IRR, X(201)), (ICASE, X(202)),
     5(IW5, X(203)), (TAB6, X(206)), (DELT2, X(231)), (DELY1, X(256)),
     6(DELY11, X(296)), (ARNO1, X(336)), (ARNO2, X(376)),
     7(ARNO3, X(416)), (SGMAX1, X(456)), (SGMAX2, X(496)),
     8(SGMAX3,X(536)), (T, X(576)), (AKSIG, X(616)), (SLOPE, X(656)),
     9(VELOS, X(696)), (WT, X(736)), (P1, X(776)), (TBLM2, X(853))
      EQUIVALENCE (P2, X(1473)), (AK1, X(1513)), (AK2, X(1553)),
     1(ABR, X(1593)), (IA, X(1633)), (M3, X(1673)), (SIG, X(1713)),
     2(AM, X(1753)), (N, X(1793)), (NEND, X(1833)), (AL6, X(1850)),
     3(AL5, X(1851)), (AL4, X(1852)), (AL3, X(1853)), (AL2, X(1854)),
     4(AL1, X(1855)), (TBLI2, X(1856)), (M5, X(3398)), (N1FLAG, X(3438))
     5,(F, X(3478)), (N6, X(3518)), (N2, X(3558)), (P, X(3598)),
     6(SCLTRB, X(3638)), (TABL1, X(3678)), (TABL2, X(3703)),
     7(TABL3, X(3728)), (TABL4, X(3753)), (TABL5, X(3778)),
     8(TABL6, X(3803)), (TABL7, X(1287)), (TABL8, X(1312)),
     9(TABL9, X(1337)), (DELT3, X(1362)), (DELT4, X(1387))
      EQUIVALENCE (DELT5, X(1412)), (DELT6, X(1437)), (L1, X(1462))
      EQUIVALENCE    (CBART,X(3837)),(AST,X(3838)),(YAW,X(3839))
     1    ,          (CYBT,X(3879)),(CYBTO,X(3919))
   10 FORMAT(1H1, 17HREFERENCE RUN NO. I6, 4X, 8HCASE NO. I6)
   20 FORMAT(1H0, 4X, 4HIEND, 1X, 4HKEND, 1X, 2HI4, 4X, 6HS-ULT., 3X,
     17H---S---, 3X, 6HC-BAR-, 3X, 10H---NEND---, 2X, 2HL1, 2X, 3HIW1,
     22X, 3HIW2, 2X, 3HIW3, 2X, 3HIW4, 2X, 3HIW5)
   25 FORMAT(1H0, 4X, 4HIEND, 1X, 4HKEND, 1X, 2HI4, 4X, 6HS-ULT., 3X,
     17H---S---, 3X, 6HC-BAR-, 3X, 10H---NEND---, 2X, 2HL1, 2X, 3HIW1,
     22X, 3HIW2, 2X,3HIW3,2X,3HIW4,2X,3HIW5,2X,*V.T.CHORD*,2X,*V.T.AREA*
     3 )
   30 FORMAT(1H , 5X, I2, 3X, I2, 2X, I2, F12.3, F8.2, 2X, F7.2,
     13X, F10.0, 3X, I1, 3X, I1, 4X, I1, 4X, I1, 4X, I1, 4X, I1)
   35 FORMAT(1H , 5X, I2, 3X, I2, 2X, I2, F12.3, F8.2, 2X, F7.2,
     13X, F10.0, 3X, I1, 3X, I1, 4X, I1, 4X, I1, 4X, I1, 4X, I1,
     2   3X,F8.2,2X,F8.2   )
   40 FORMAT(1H0, 2X, 4HSEG., 1X,2HM3, 2X, 2HM5, 2X, 6HISTRES, 2X, 2HIA,
     12X, 6HN1FLAG, 1X, 13H------P------, 2X, 2HN6, 2X, 2HN2, 2X,
```

```
    213H-----AM------, 3X, 17H--------F--------, 10X, 8HDELTA Y1,
    310X, 9HDELTA Y11)
 50 FORMAT(1H , 3X, I2, 2X, I2, 2X, I2, 4X, I2, 4X, I2, 4X, I1, 4X,
    1F13.0, 2X, I2, 2X, I2, 2X, F13.0, 3X, F17.4, 3X, F17.4, 3X, F17.4)
 60 FORMAT(1H0, 4HSEG., 1X, 14H---N SUB 01---,2X,14H---N SUB 02---,2X
    1, 14H---N SUB 03---, 2X, 11H--SIG DY1--, 2X, 11H--SIG DY2--, 2X,
    211H--SIG DY3--)
 70 FORMAT(1H , 1X, I2, 2X, F14.4, 2X, F14.4, 2X, F14.4, 2X, F11.4,
    12X, F11.4, 2X, F11.4)
 80 FORMAT(1H0, 2X,4HSEG.,3X,6HKSIGMA, 3X, 5HSLOPE, 5X, 2HVE, 4X,
    110H-----W----, 6X, 2HP1, 10X, 2HP2, 9X, 2HB1, 7X, 2HB2, 10X,
    25HA-BAR, 5X, 14H------T-------, 4X, 1HN)
 90 FORMAT(1H , 3X, I2, 3X, F8.5, 1X, F6.2, 2X, F7.2, 2X, F10.0, 2X,
    1F10.6, 2X, F10.6, 2X, F7.3, 2X, F7.3, F16.6, 2X, F14.3, 3X, I2)
100 FORMAT(1H0, 3X,4HSEG.,4X,17HAIR DENSITY RATIO, 5X,
    119HSCALE OF TURBULENCE)
105 FORMAT(1H0, 3X,4HSEG.,4X,17HAIR DENSITY RATIO, 5X,
    119HSCALE OF TURBULENCE ,11X,*SFC - T *, 5X, * SFC - TO*, 5X,
    2*     IYAW     * )
110 FORMAT(1H , 4X, I2, 9X, F9.5, 14X, F10.3)
115 FORMAT(1H , 4X, I2,9X,F9.5,14X,F10.3,14X,F8.4, 6X,F8.4,8X,F16.0 )
120 FORMAT(1H0, 4X, 2HLL, 4X, 11HSTRESS TBL1, 2X, 11HSTRESS TBL2, 2X,
    111HSTRESS TBL3, 2X, 11HSTRESS TBL4, 2X, 11HSTRESS TBL5, 2X,
    211HSTRESS TBL6, 2X, 11HSTRESS TBL7, 2X, 11HSTRESS TBL8)
130 FORMAT(1H )
140 FORMAT(1H , 4X, I2, 3X, F13.2, 1X, F13.2, F13.2, F13.2, F13.2,
    1F13.2, F13.2, F13.2)
150 FORMAT(1H , 4X, I2, 3X, F13.2, 1X, F13.2,F13.2,F13.2,F13.2,F13.2)
160 FORMAT(1H , 4X, 2HLL, 4X, 16HDELTA Y--TABLE 1, 2X,
    116HDELTA Y--TABLE 2, 2X, 16HDELTA Y--TABLE 3, 2X,
    216HDELTA Y--TABLE 4, 2X, 16HDELTA Y--TABLE 5, 2X,
    316HDELTA Y--TABLE 6)
170 FORMAT(1H , 4X, I2, F18.3, F18.3, F18.3, F18.3, F18.3, F18.3)
180 FORMAT(1H , 4X, 2HLL, 4X, 16HCUM CYCLES TBL 1, 2X,
    116HCUM CYCLES TBL 2, 2X, 16HCUM CYCLES TBL 3, 2X,
    216HCUM CYCLES TBL 4, 2X, 16HCUM CYCLES TBL 5, 2X,
    316HCUM CYCLES TBL 6)
190 FORMAT(1H , 4X, I2, F19.3, F18.3, F18.3, F18.3, F18.3, F18.3)
200 FORMAT(1H0, 4X, 2HLL, 4X, 11HSTRESS TBL9, 2X, 12HSTRESS TBL10, 1X,
    112HSTRESS TBL11, 1X, 12HSTRESS TBL12, 1X, 12HSTRESS TBL13, 1X,
    212HSTRESS TBL14, 1X, 12HSTRESS TBL15, 1X, 12HSTRESS TBL16)
210 FORMAT(1H , 4X, 2HLL, 4X, 13HMAX STRESS(1),2X,13HMIN STRESS(1),6X,
    19HCYCLES(1), 5X, 13HMAX STRESS(2), 2X, 13HMIN STRESS(2), 6X,
    29HCYCLES(2))
220 FORMAT(1H , 4X, I2, F17.0, F15.0, F18.0, F15.0, F15.0, F18.0)
230 FORMAT(1H , 4X, 2HLL, 4X, 13HMAX STRESS(3),2X,13HMIN STRESS(3),6X,
    19HCYCLES(3))
240 FORMAT(1H , 4X, I2, F17.0, F15.0, F18.0)
250 FORMAT(1H0, 2X, 11HS-N TABLE = I2, 2X, 14HIA AND/OR I4 = I2)
260 FORMAT(1H , 4X, 18HNO. OF Y ENTRIES = F4.0, 4X,
    118HNO. OF X ENTRIES = F4.0,2X,23HMAX CYCLES TO FAILURE = F12.0)
270 FORMAT(1H0, 7X, 1HY, 12X, 1HX, 10X, 4HY1,X, 9X, 4HY2,X, 9X,
    14HY3,X, 9X, 4HY4,X, 9X, 4HY5,X, 9X, 4HY6,X, 9X, 4HY7,X)
```

```
280 FORMAT(1H , F12.3, F14.4, F13.0, F13.0, F13.0,
   1F13.0, F13.0, F13.0, F13.0)
290 FORMAT(1H0, 8X, 4HY8,X, 10X, 4HY9,X, 9X, 5HY10,X, 9X, 5HY11,X,
   19X, 5HY12,X, 9X, 5HY13,X, 9X, 5HY14,X, 9X, 5HY15,X)
300 FORMAT(1H , 4X, F12.0, 2X, F12.0, 2X, F12.0, 2X, F12.0, 2X, F12.0,
   12X, F12.0, 2X, F12.0, 2X, F12.0)
310 FORMAT(1H0, 32HTHE FOLLOWING DATA IS INPUT DATA)
320 FORMAT(1H , 51HSTRESS TABLES, S = X = F(Y). LL 1 = NO. OF Y ENTRIE
   143HS. LL 2-16 = Y VALUES. LL 17-31 = X VALUES.)
    WRITE(6,310)
    DO 323  I=1,IEND
    IF(M3(I).NE.14.AND.M3(I).NE.15) GO TO 325
323 CONTINUE
    WRITE(6,25)
    WRITE(6,35) IEND,KEND,I4,SIGULT,WAREA,AC, NEND, L1, IW1, IW2 ,
   1  IW3, IW4, IW5, CBART, AST
    GO TO 327
325 WRITE(6,20)
    WRITE(6,30) IEND, KEND, I4, SIGULT, WAREA, AC, NEND, L1, IW1, IW2,
   1 IW3, IW4, IW5
327 WRITE(6,40)
    WRITE(6,50) (I, M3(I), M5(I), ISTRES(I), IA(I), N1FLAG(I), P(I),
   1 N6(I), N2(I), AM(I), F(I), DELY1(I), DELY11(I), I = 1, IEND)
    WRITE(6,10) IRR, ICASE
    WRITE(6,310)
    WRITE(6,60)
    WRITE(6,70)(I,ARNO1(I),ARNO2(I), ARNO3(I), SGMAX1(I), SGMAX2(I),
   1 SGMAX3(I), I = 1, IEND)
    WRITE(6,10) IRR, ICASE
    WRITE(6,310)
    WRITE(6,80)
    WRITE(6,90)(I,AKSIG(I),SLOPE(I), VELOS(I), WT(I), P1(I), P2(I),   A
   1K1(I), AK2(I), ABR(I), T(I), N(I), I = 1, IEND)
    DO 330 I = 1, IEND
    IF(M3(I).EQ.14 .OR. M3(I) .EQ. 15 ) GO TO 340
    IF(M3(I) .EQ. 13) GO TO 340
330 CONTINUE
    GO TO 350
340 WRITE(6,10) IRR, ICASE
    WRITE(6,310)
    DO 343  I=1,IEND
    IF(M3(I).NE.14.AND.M3(I).NE.15) GO TO 345
343 CONTINUE
    WRITE(6,105)
    WRITE(6,115)(I,SIG(I),SCLTRB(I),CYBT(I),CYBTO(I),YAW(I),I=1,IEND)
    GO TO  350
345 WRITE(6,100)
    WRITE(6,110) (I, SIG(I), SCLTRB(I), I = 1, IEND)
350 DO 360 I = 1, IEND
    IF((M5(I) .GT. 0) .AND. (M5(I) .LT. 7)) GO TO 370
360 CONTINUE
    GO TO 380
370 WRITE(6,10) IRR, ICASE
```

```
      WRITE(6,310)
      WRITE(6,130)
      WRITE(6,160)
      WRITE(6,170) (J, DELT1(J), DELT2(J), DELT3(J), DELT4(J), DELT5(J),
     1DELT6(J), J = 1, 25)
  380 DO 390 I = 1, IEND
      IF((M3(I) .GT. 0) .ANC. (M3(I) .LT. 7)) GC TC 400
  390 CCNTINUE
      GC TO 410
  400 WRITE(6,10) IRR, ICASE
      WRITE(6,310)
      WRITE(6,130)
      WRITE(6,180)
      WRITE(6,190) (J, TAB1(J), TAB2(J), TAB3(J), TAB4(J), TAB5(J),
     1TAB6(J), J = 1, 25)
  410 DO 420 I = 1, IENC
      IF((M3(I) .GT. 9) .ANC. (M3(I) .LT. 12)) GC TC 430
  420 CCNTINUE
      GO TO 440
  430 WRITE(6,10) IRR, ICASE
      WRITE(6,310)
      WRITE(6,130)
      WRITE(6,210)
      WRITE(6,220) (J, TABL1(J), TABL2(J), TABL3(J), TABL4(J), TABL5(J),
     1 TABL6(J), J = 1, 25)
  440 DO 450 I = 1, IEND
      IF(M3(I) .EQ. 12) GO TO 46C
  450 CONTINUE
      GC TO 470
  460 WRITE(6,10) IRR, ICASE
      WRITE(6,310)
      WRITE(6,130)
      WRITE(6,230)
      WRITE(6,240) (J, TABL7(J), TABL8(J), TABL9(J), J = 1, 25)
  470 DO 480 I = 1, IEND
      IF((ISTRES(I) .GT. 0) .ANC. (ISTRES(I) .LT. 9)) GO TC 490
  480 CCNTINUE
      GC TO 500
  490 WRITE(6,10) IRR, ICASE
      WRITE(6,310)
      WRITE(6,130)
      WRITE(6,320)
      WRITE(6,120)
      WRITE(6,140)(J, TBLM2(J), TBLM2(J+31), TBLM2(J+62), TBLM2(J+93),
     1 TBLM2(J+124), TBLM2(J+155),TBLM2(J+186),TBLM2(J+217),J = 1,31)
  500 DO 510 I = 1, IENC
      IF((ISTRES(I) .GT. 8) .AND. (ISTRES(I) .LT. 15)) GC TO 520
  510 CCNTINUE
      GC TO 530
  520 WRITE(6,10) IRR, ICASE
      WRITE(6,310)
      WRITE(6,130)
      WRITE(6,320)
```

```
      WRITE(6,200)
      WRITE(6,150) (J, TBLM2(J+248), TBLM2(J+279), TBLM2(J+310),
     1 TBLM2(J+341), TBLM2(J+372), TBLM2(J+403), J = 1, 31)
  530 A5 = 0.0
      B5 = 0.0
      C5 = 0.0
      D5 = 0.0
      E5 = 0.0
      F5 = 0.0
      G5 = 0.0
  540 DO 620 I = 1, IEND
      IF(IA(I) .LT. 7) I2 = IA(I)
      IF((IA(I) .GT. 6) .AND. (IA(I) .LT. 13)) I2 = (IA(I) - 6)
      IF((IA(I) .GT. 12) .AND. (IA(I) .LT. 19)) I2 = (IA(I) - 12)
      IF(IA(I) .GT. 18) I2 = (IA(I) - 18)
      ICALL = I2
      GO TO(550,560,570,580,590,600), ICALL
  550 IF(A5.EQ. 1.0) GO TO 610
      WRITE(6,10) IRR, ICASE
      WRITE(6,310)
      WRITE(6,250)  ICALL, IA(I)
      WRITE(6,260) TBLI2(1), TBLI2(17), AL1
      WRITE(6,270)
      WRITE(6,280) (TBLI2(J), TBLI2(J + 16), TBLI2(J + 31),
     1TBLI2(J + 46), TBLI2(J + 61), TBLI2(J + 76), TBLI2(J + 91),
     2TBLI2(J + 106), TBLI2(J + 121), J = 2, 16)
      WRITE(6,290)
      WRITE(6,300) (TBLI2(J + 136), TBLI2(J + 151), TBLI2(J + 166),
     1TBLI2(J + 181), TBLI2(J + 196), TBLI2(J + 211), TBLI2(J + 226),
     2TBLI2(J + 241), J = 2, 16)
      A5= 1.0
      GO TO 610
  560 IF(B5.EQ. 1.0) GO TO 610
      WRITE(6,10) IRR, ICASE
      WRITE(6,310)
      WRITE(6,250) ICALL, IA(I)
      WRITE(6,260) TBLI2(258), TBLI2(274), AL2
      WRITE(6,270)
      WRITE(6,280) (TBLI2(J), TBLI2(J + 16), TBLI2(J + 31),
     1TBLI2(J + 46), TBLI2(J + 61), TBLI2(J + 76), TBLI2(J + 91),
     2TBLI2(J + 106), TBLI2(J + 121), J = 259, 273)
      WRITE(6,290)
      WRITE(6,300) (TBLI2(J + 136), TBLI2(J + 151), TBLI2(J + 166),
     1TBLI2(J + 181), TBLI2(J + 196), TBLI2(J + 211), TBLI2(J + 226),
     2TBLI2(J + 241), J = 259, 273)
      B5= 1.0
      GO TO 610
  570 IF(C5.EQ. 1.0) GO TO 610
      WRITE(6,10) IRR, ICASE
      WRITE(6,310)
      WRITE(6,250) ICALL, IA(I)
      WRITE(6,260) TBLI2(515), TBLI2(531), AL3
      WRITE(6,270)
```

```
      WRITE(6,280) (TBLI2(J), TBLI2(J + 16), TBLI2(J + 31),
     1TBLI2(J + 46), TBLI2(J + 61), TBLI2(J + 76), TBLI2(J + 91),
     2TBLI2(J + 106), TBLI2(J + 121), J = 516, 530)
      WRITE(6,290)
      WRITE(6,300) (TBLI2(J + 136), TBLI2(J + 151), TBLI2(J + 166),
     1TBLI2(J + 181), TBLI2(J + 196), TBLI2(J + 211), TBLI2(J + 226),
     2TBLI2(J + 241), J = 516, 530)
      C5= 1.0
      GO TO 610
  580 IF(D5.EQ. 1.0) GO TO 610
      WRITE(6,10) IRR, ICASE
      WRITE(6,310)
      WRITE(6,250) ICALL, IA(I)
      WRITE(6,260) TBLI2(772), TBLI2(788), AL4
      WRITE(6,270)
      WRITE(6,280) (TBLI2(J), TBLI2(J + 16), TBLI2(J + 31),
     1TBLI2(J + 46), TBLI2(J + 61), TBLI2(J + 76), TBLI2(J + 91),
     2TBLI2(J + 106), TBLI2(J + 121), J = 773, 787)
      WRITE(6,290)
      WRITE(6,300) (TBLI2(J + 136), TBLI2(J + 151), TBLI2(J + 166),
     1TBLI2(J + 181), TBLI2(J + 196), TBLI2(J + 211), TBLI2(J + 226),
     2TBLI2(J + 241), J = 773, 787)
      D5= 1.0
      GO TO 610
  590 IF(E5.EQ. 1.0) GO TO 610
      WRITE(6,10) IRR, ICASE
      WRITE(6,310)
      WRITE(6,250) ICALL, IA(I)
      WRITE(6,260) TBLI2(1029), TBLI2(1045), AL5
      WRITE(6,270)
      WRITE(6,280) (TBLI2(J), TBLI2(J + 16), TBLI2(J + 31),
     1TBLI2(J + 46), TBLI2(J + 61), TBLI2(J + 76), TBLI2(J + 91),
     2TBLI2(J + 106), TBLI2(J + 121), J = 1030, 1044)
      WRITE(6,290)
      WRITE(6,300) (TBLI2(J + 136), TBLI2(J + 151), TBLI2(J + 166),
     1TBLI2(J + 181), TBLI2(J + 196), TBLI2(J + 211), TBLI2(J + 226),
     2TBLI2(J + 241), J = 1030, 1044)
      E5= 1.0
      GO TO 610
  600 IF(F5.EQ. 1.0) GO TO 610
      WRITE(6,10) IRR, ICASE
      WRITE(6,310)
      WRITE(6,250) ICALL, IA(I)
      WRITE(6,260) TBLI2(1286), TBLI2(1302), AL6
      WRITE(6,270)
      WRITE(6,280) (TBLI2(J), TBLI2(J + 16), TBLI2(J + 31),
     1TBLI2(J + 46), TBLI2(J + 61), TBLI2(J + 76), TBLI2(J + 91),
     2TBLI2(J + 106), TBLI2(J + 121), J = 1287, 1301)
      WRITE(6,290)
      WRITE(6,300) (TBLI2(J + 136), TBLI2(J + 151), TBLI2(J + 166),
     1TBLI2(J + 181), TBLI2(J + 196), TBLI2(J + 211), TBLI2(J + 226),
     2TBLI2(J + 241), J = 1287, 1301)
      F5= 1.0
```

```
610 IF(G5 .EQ. 1.0) GO TO 630
620 CONTINUE
    IF(I4 .EQ. 0) GO TO 630
    IA(1) = I4
    G5 = 1.0
    GO TO 540
630 RETURN
    END
```

```
      SUBRCUTINE ONEVAR(ARGUMT, TABLE, OUTPUT, NSEGNM)
C
C CNEVAR IS AN INTERPOLATION RCUTINE - ONE FUNCTICN OF CNE
C VARIABLE, I.E. X=F(Y) - LINEAR CR QUADRATIC.
C   ARGUMENTS CF THE SUBRCUTINE ARE AS FOLLCWS
C     ARGUMT   = INPUT INTERPOLATICN ARGUMENT (Y)
C     NXDIR    = TYPE OF INTERP. 1 FCR LINEAR, 2 FOR QUAD.
C     TABLE    = SET OF Y VALUES FOLLOWED BY THE X VALUES
C     OUTPUT   = INTERPOLATED VALUE CF X = F(Y)
C     NER      = ERROR COCE
C                1 = OK, INTERPCLATICN SUCCESSFUL.
C                2 = OFF CHART LOW END. MIN. VAL. SUBSTITUTED
C                3 = OFF CHART HIGH END. MAX.VAL. SUBSTITUTED
C                4 = NC.OF X ENTRIES IS NCT 2 TO 15 (IF NXDIR
C                    IS 1). CR, IT IS NOT 3 TO 15 (IF NXDIR
C                    IS 2).
C                5 = Y ENTRIES NOT IN ASCENDING ORDER.
      DIMENSION TABLE(31)
C     DIMENSION TABLE(31)
      NCENTR = TABLE(1) + 0.5
      NXCIR = 1
      NER = 0
      IF (NXDIR-1)10,10,20
   10 IF (NCENTR-2)40,30,30
   20 IF (NCENTR-3)40,30,30
   30 IF (15- NOENTR)40,50,50
   40 NER = 4
      GO TO 260
   50 IF((ARGUMT + ARGUMT * 0.001) - TABLE(2))60,70,90
   60 NER = 2
      GC TO 80
   70 NER = 1
   80 ICUT = 2
      GO TO 250
   90 IF (TABLE(NOENTR+1)-ARGUMT)100,110,140
  100 NER = 3
      GC TO 120
  110 NER = 1
  120 IOUT = NOENTR+1
  130 GC TC 250
C AT THIS STAGE OF THE GAME, ERRCR CCNDITIONS 2,3,4 HAVE
C BEEN TESTED FOR AND HAVE BEEN PASSED.
  140 DC 240JK=1, NOENTR
      I = JK
      IF (I-1)170,170,150
  150 IF (TABLE(I+1)-TABLE(I))160,160,170
  160 NER = 5
      GO TO 260
  170 IF (TABLE(I+1)-ARGUMT)240,180,190
  180 NER = 1
      ICUT = I+1
```

```
      GO TO 250
  190 IF (NXDIR-1)200,200,210
  200 NER = 1
      OUTPUT = TABLE(I+15)+ (ARGUMT - TABLE(I))   * (TABLE
     1(I+16) - TABLE(I+15))/(TABLE(I+1)-TABLE(I))
      GO TO 360
  210 IF (NCENTR-I)230,220,230
  220 I = I-1
  230 OUTPUT= (TABLE(I+15) * (ARGUMT-TABLE(I+1)) * (ARGUMT -
     1TABLE(I+2)))/ ((TABLE(I) - TABLE(I+1)) * (TABLE(I)    -
     2TABLE(I+2))) + (TABLE(I+16) * (ARGUMT - TABLE(I))   * (
     3ARGUMT - TABLE(I+2))) / ((TABLE(I+1)-TABLE(I))    * (
     4TABLE(I+1)-TABLE(I+2))) + (TABLE(I+17) * (ARGUMT -
     5TABLE(I))    * (ARGUMT - TABLE(I+1)))/((TABLE(I+2)-
     6TABLE(I))    *(TABLE(I+2) - TABLE(I+1)))
      NER = 1
      GO TO 360
  240 CONTINUE
  250 OUTPUT = TABLE(IOUT+15)
  260 IF (NER-2)360,270,290
  270 WRITE(6,280) ARGUMT, NSEGNM
  280 FORMAT (1H0, 38HONEVAR INTERPOLATION ERROR. Y IS TOO S
     19HMALL. Y = E14.6, 4X, 6HSEG. = I2)
      GO TO 360
  290 IF (NER-4)300,320,340
  300 WRITE(6,310) ARGUMT, NSEGNM
  310 FORMAT (1H0, 38HONEVAR INTERPOLATION ERROR. Y IS TOO L
     19HARGE. Y = E14.6, 4X, 6HSEG. = I2)
      GO TO 360
  320 WRITE (6,330)
  330 FORMAT (1H0,39HONEVAR INTERP. ERROR. THE NO. OF Y ENTR
     137HIES IS EITHER TOO SMALL OR TOO LARGE., 2X, 6HSEG. = I2)
      GO TO 360
  340 WRITE (6,350)
  350 FORMAT (1H0,39HONEVAR INTERPOLATION ERROR. Y ENTRIES A
     126HRE NOT IN ASCENDING ORDER. , 2X, 6HSEG. = I2)
  360 RETURN
      END
```

```
      SUBROUTINE TWOVIN(XARG, YARG, TABLE, CUTPUT, NSEG, LEVEL)
C
C    ARGUMENTS OF THE SUBROUTINE ARE AS FOLLOWS
C      XARG   = INPUT INTERPCLATION ARGUMENT (X)
C      YARG   = INPUT INTERPOLATICN ARGUMENT (Y)
C      TABLE  = SET OF VALUES. SEE THE DESCRIPTION CF THIS
C      OUTPUT  = INTERPOLATED VALUE OF Z = F(X,Y)
      DIMENSION TABLE(257)
   10 FORMAT(1H0, 37HSN INTERP. ERROR. Y IS TOC SMALL. Y = E14.6, 2X,
     15HSEG = I3, 2X, 12HLOAD LEVEL = I3)
   20 FORMAT(1H0, 37HSN INTERP. ERRCR. Y IS TOC LARGE. Y = E14.6, 2X,
     15HSEG = I3, 2X, 12HLOAD LEVEL = I3)
   30 FORMAT(1H0, 37HSN INTERP. ERROR. X IS TOO LARGE. X = E14.6, 2X,
     15HSEG = I3, 2X, 12HLOAD LEVEL = I3)
   40 FCRMAT(1H0, 37HSN INTERP. ERROR. X IS TOO LARGE. X = E14.6, 2X,
     119HY IS TOO SMALL. Y = E14.6, 2X,5HSEG = I3,2X,12HLOAD LEVEL = I3)
   50 FORMAT(1H0, 37HSN INTERP. ERROR. X IS TOC LARGE. X = E14.6, 2X,
     119HY IS TOO LARGE. Y = E14.6, 2X,5HSEG = I3,2X,12HLOAD LEVEL = I3)
   60 FORMAT(1H0, 37HSN INTERP. ERRCR. Y IS TOC SMALL. Y = E14.6, 2X,
     13HX = E14.6, 2X, 13H(GAG SEGMENT))
   70 FCRMAT(1H0, 37HSN INTERP. ERRCR. Y IS TCC LARGE. Y = E14.6, 2X,
     13HX = E14.6, 2X, 13H(GAG SEGMENT))
   80 FORMAT(1H0, 37HSN INTERP. ERRCR. X IS TOO LARGE. X = E14.6, 2X,
     13HY = E14.6, 2X, 13H(GAG SEGMENT))
   90 FCRMAT(1H0, 37HSN INTERP. ERROR. X IS TOC LARGE. X = E14.6, 2X,
     119HY IS TOO SMALL. Y = E14.6, 2X, 13H(GAG SEGMENT))
  100 FCRMAT(1H0, 37HSN INTERP. ERROR. X IS TCC LARGE. X = E14.6, 2X,
     119HY IS TOO LARGE. Y = E14.6, 2X, 13H(GAG SEGMENT))
  110 FORMAT(1H0,45HGUST ALLEV. INTRP. ERROR. Y IS TOC SMALL. Y = E14.6,
     12X, 5HSEG = I3)
  120 FORMAT(1H0,45HGUST ALLEV. INTRP. ERRCR. Y IS TCC LARGE. Y = E14.6,
     12X, 5HSEG = I3)
  130 FCRMAT(1H0,45HGUST ALLEV. INTRP. ERRCR. X IS TOC LARGE. X = E14.6,
     12X, 5HSEG = I3)
  140 FORMAT(1H0,45HGUST ALLEV. INTRP. ERROR. X IS TOC LARGE. X = E14.6,
     12X, 19HY IS TOO SMALL. Y = E14.6, 1X, 5HSEG = I3)
  150 FCRMAT(1H0,45HGUST ALLEV. INTRP. ERROR. X IS TOC LARGE, X = E14.6,
     12X, 19HY IS TOO LARGE. Y = E14.6, 1X, 5HSEG = I3)
      J5 = 0
      K  = 0
      NER = 0
      NER2= 0
      NER5= 0
      NXDIR = 2
      NXENTR = TABLE(17) + 0.5
      NYENTR = TABLE(1) + 0.5
  160 DC 300 JK = 1, NYENTR
      IY=JK
      IF (IY-1)230,230,170
  170 IF(IY - NYENTR)180,200,200
  180 IF(TABLE(IY+1) - TABLE(IY))190,190,260
```

```
190 NER = 9
    GO TO 740
200 IF(TABLE(IY + 1) - YARG)210,220,310
210 NER = 3
    NER2 = 13
220 J5 = 1
    IY = IY + 1
    GO TO 310
230 IF(TABLE(IY + 1) - YARG)270,250,240
240 NER = 2
    NER2 = 12
250 IY = IY + 1
    J5 = 1
    GO TO 310
260 IF(TABLE(IY + 1) - YARG)280,310,310
270 CYINT = ((YARG - TABLE(IY + 1)) / (TABLE(IY + 2) - TABLE(IY + 1)))
    IF(CYINT - 0.001)250,300,300
280 CYINT = ((YARG - TABLE(IY + 1)) / (TABLE(IY + 1) - TABLE(IY)))
290 IF(CYINT - 0.001)310,300,300
300 CONTINUE
310 DO 460 JK = 1, NXENTR
    IX=JK
    IF(IX - 1)420,420,320
320 IF(IX - NXENTR)330,390,390
330 IF(TABLE(IX + 17) - TABLE(IX + 16))340,340,450
340 NER = 9
    GO TO 740
350 NER = 5
    NER5= 13
360 IX = IX + 1
    IF(J5.EQ. 1) GO TO 470
    GO TO 480
370 CKINT =ABS((TABLE(IX+17) - XARG) / (TABLE(IX+18) - TABLE(IX+17)))
380 IF(CKINT - 0.50)480,480,460
390 IF(TABLE(IX+17) - XARG)350,360,400
400 CKINT =ABS((TABLE(IX+17) - XARG)/(TABLE(IX+17)-TABLE(IX+16)))
    IF(CKINT - 0.50)410,410,350
410 IX = IX - 1
    GO TO 480
420 IF(TABLE(IX+17) - XARG)460,460,430
430 NER = 4
440 IX = IX + 1
    GO TO 480
450 IF(TABLE(IX+17) - XARG)370,370,480
460 CONTINUE
470 IN = 15 * IY + 1 + IX
    OUTPUT = ALOG10(TABLE(IN))
    GO TO 710
480 IN = 15 * IY + 1 + IX
    AN1 = ALOG10(TABLE(IN))
    IBOUND = (NXENTR - 1)
    IF(IX .GT. NXENTR) GO TO 510
    AN2 = ALOG10(TABLE(IN + 1))
```

```
      IF((IX .EQ. NXENTR) .AND. (J5 .EQ. 1)) GO TO 500
      IF((IX .EQ. NXENTR) .AND. (J5 .EQ. 0)) GO TO 490
      AN3 = ALOG10(TABLE(IN + 2))
      IF(IY .GT. NYENTR) GO TO 500
      AN5 = ALOG10(TABLE(IN + 17))
      AN6 = ALOG10(TABLE(IN + 16))
      IF(IX .EQ. IBOUND) GO TO 510
      AN4 = ALOG10(TABLE(IN + 18))
      GO TO 510
  490 AN6 = ALOG10(TABLE(IN + 16))
      GO TO 510
  500 AN4 = AN2
      AN5 = AN2
      AN6 = AN1
  510 IF(NXDIR - 1)520,520,600
  520 IF(J5- 1)540,530,540
  530 BX= 0.0
      GO TO 550
  540 BX= ((YARG - TABLE(IY)) / (TABLE(IY+1) - TABLE(IY)))
  550 IF(IX - NXENTR)570,570,560
  560 OUTPUT = ALOG10(TABLE(IN)) + BX* (ALOG10(TABLE(IN+15)) -
     1ALOG10(TABLE(IN)))
      GO TO 710
  570 XARGMX = TABLE(NXENTR + 17)
      IF(TABLE(IX+17) - XARGMX)590,580,580
  580 TABLE(IX + 18) = TABLE(IX + 17)
      TABLE(IN + 17) = TABLE(IN + 16)
  590 CX= XARG - (TABLE(IX+16) + BX* (TABLE(IX+17) - TABLE(IX+16)))
      DX= ALOG10(TABLE(IN+1)) + BX* (ALOG10(TABLE(IN+17)) -
     1ALOG10(TABLE(IN+1)))
      EX= ALOG10(TABLE(IN)) + BX* (ALOG10(TABLE(IN+16)) -
     1ALOG10(TABLE(IN)))
      FX= TABLE(IX+17) - TABLE(IX+16) + BX* (TABLE(IX+18) -
     12.0 * (TABLE(IX+17)) + TABLE(IX+16))
      IF(FX .EQ. 0.0) FX= 1.0
      OUTPUT = ALOG10(TABLE(IN)) + BX* (ALOG10(TABLE(IN+16)) -
     1ALOG10(TABLE(IN))) + (((CX) * (DX - EX)) / (FX))
      GO TO 710
  600 IF(J5- 1)620,610,620
  610 BX= 0.0
      GO TO 630
  620 BX= ((YARG - TABLE(IY)) / (TABLE(IY+1) - TABLE(IY)))
  630 IF(IX - NXENTR)650,650,640
  640 AN7 = ALOG10(TABLE(IN + 15))
      OUTPUT = AN1 + BX* (AN7 - AN1)
      GO TO 710
  650 IBOUND = NXENTR - 1
      IF(IX .EQ. IBOUND) GO TO 670
      XARGMX = TABLE(NXENTR + 17)
      IF(TABLE(IX+17) - XARGMX)680,660,660
  660 TABLE(IX + 18) = TABLE(IX + 17)
      AN3 = AN2
      AN5 = AN2
```

```
670 TABLE(IX + 19) = TABLE(IX + 17)
    IF(IX .EQ. IBOUND) TABLE(IX + 19) = TABLE(IX + 18)
    AN4 = AN5
680 CX= TABLE(IX + 18) + BX* (TABLE(IX+19) - TABLE(IX+18))
    DX= TABLE(IX + 17) + BX* (TABLE(IX+18) - TABLE(IX+17))
    EX= TABLE(IX + 16) + BX* (TABLE(IX+17) - TABLE(IX+16))
    FX= AN3 + BX* (AN4 - AN3)
    GX= AN2 + BX* (AN5 - AN2)
    HX= AN1 + BX* (AN6 - AN1)
    IF((IX .GE. IBOUND) .AND.  ((BX * 1.001) .GE. 1.0)) GO TO 700
690 OUTPUT = (HX*(((XARG - DX)*(XARG - CX))/((EX - DX)*(EX - CX))))
   1+ (GX*(((XARG - EX)*(XARG - CX))/((DX - EX)*(DX - CX))))
   2+ (FX*(((XARG - EX)*(XARG - DX))/((CX - EX)*(CX - DX))))
    GO TO 710
700 OUTPUT = GX + (HX - GX) * ((DX - XARG) / (DX - EX))
710 IF(NSEG .EQ. 50) GO TO 720
    IF(NSEG .GT. 50) GO TO 730
    IF(NER .EQ. 2) WRITE(6,10) YARG, NSEG, LEVEL
    IF(NER .EQ. 3) WRITE(6,20) YARG, NSEG, LEVEL
    IF((NER .EQ. 5) .AND. (NER2 .EQ. 0))WRITE(6,30) XARG, NSEG, LEVEL
    IF((NER2 .EQ. 12).AND.(NER5 .EQ. 13)) WRITE(6,40) XARG, YARG, NSEG
   1,LEVEL
    IF((NER2 .EQ. 13).AND.(NER5 .EQ. 13)) WRITE(6,50) XARG, YARG, NSEG
   1,LEVEL
    GO TO 760
720 IF(NER .EQ. 2) WRITE(6,60) YARG, XARG
    IF(NER .EQ. 3) WRITE(6,70) YARG, XARG
    IF((NER .EQ. 5) .AND. (NER2 .EQ. 0))WRITE(6,80) XARG, NSEG, LEVEL
    IF((NER2 .EQ. 12) .AND. (NER5 .EQ. 13)) WRITE(6,90) XARG, YARG
    IF((NER2 .EQ. 13) .AND. (NER5 .EQ. 13)) WRITE(6,100) XARG, YARG
    GO TO 760
730 NSEG = NSEG - 50
    IF(NER .EQ. 2) WRITE(6,110) YARG, NSEG
    IF(NER .EQ. 3) WRITE(6,120) YARG, NSEG
    IF((NER .EQ. 5) .AND. (NER2 .EQ. 0))WRITE(6,130) XARG, NSEG
    IF((NER2 .EQ. 12) .AND. (NER5 .EQ. 13)) WRITE(6,140) XARG, YARG,
   1NSEG
    IF((NER2 .EQ. 13) .AND. (NER5 .EQ. 13)) WRITE(6,150) XARG, YARG,
   1NSEG
    GO TO 760
740 WRITE (6,750)
750 FORMAT (1H0,39HTWOVIN INTERPOLATION ERROR. EITHER THE
   146HX OR THE Y ENTRIES ARE NOT IN ASCENDING ORDER.)
760 RETURN
    END
```

```
      SUBROUTINE RCMAIN
C**********   MAIN PROGRAM FOR THE SPECTRUM LOADING SEQUENCE GENERATION **
C**********   PROGRAM.  IT READS AND PRINTS THE TITLE AND SOME OF THE    **
C**********   CONSTANTS USED IN THE PROGRAM.                            **
C**********   NOTE - THE SIZE OF THE DYNAMIC WORKING ARRAY IS SET BY THE
C**********          DIMENSION OF A AND BY NSIZE.  THROUGHOUT THE SPECTRUM
C**********          LOADING SEQUENCE GENERATION PROGRAM CORE ALLOCATION
C**********          IS DONE THROUGH IMPLIED EQUIVALENCES.
C**********   SUBROUTINES CALLED - ERROR, INPUTF, NEWPG                 **
      DIMENSION A(28000),N(1)
      EQUIVALENCE (A(1),N(1))
      COMMON NPG,TITLE(20),NSIZE,LEFT,IERR,IRS,IUIL,NPI,NRAN,
     1       IVP,IFI,NXY,CLIP,CLIV,FACTOR,ELIMP,ELIMV,IPFS,NPSS,IPTF,
     2       IAFS,NEXT,NOW,IFRS,MAXHP,KF,KC,A
      DATA BLANK/4H    /
      NSIZE=28000
      NEXT = 1
      NOW = 1
      DO 3 I=1,20
    3 TITLE(I)=BLANK
      READ(5,1001) (TITLE(I),I=1,20)
 1001 FORMAT (20A4)
      NPG=1
      CALL NEWPG
      IERR=0
      READ (5,*) NFT,IRS,IFRS,IUIL,IPI,IRAN,KVP,IFI,NXY,IPFS,NPSS,IPTF,
     1            IAFS,MAXHP,KF,KC
      WRITE(6,1) NFT,IRS,IFRS,IUIL,IPI,IRAN,KVP,IFI,NXY,IPFS,NPSS,IPTF,
     1            IAFS,MAXHP
    1 FORMAT (
     X 15X,*NUMBER OF FLIGHT TYPES IS.........................*,I5/
     1 15X,*VALLEY/PEAK COUPLING IS..( 1=A6PA, 2=RANDOM )....*,I5/
     F 15X,*FLIGHT SEQUENCE IS..(0=RANDOM.....................*,I5/
     F 15X,*...............(N=INPUT FLIGHT SEQUENCE, N SETS)*,/
     2 15X,*FORTRAN UNIT NUMBER IS............................*,I5/
     4 15X,*NUMBER OF INPUT PEAK LEVELS IS....................*,I5/
     5 15X,*NUMBER OF INPUT RANGE LEVELS IS...................*,I5/
     3 15X,*NUMBER OF INPUT VALLEY/PEAK RATIOS IS.............*,I5/
     6 15X,*SAVE SPECTRUM ON MAGNETIC TAPE....................*,I5/
     6 15X,*............................( 0=NO, 3=YES )...*,/
     7 15X,*NUMBER OF POINTS ON THE RANGE VS R INPUT CURVE IS*,I5/
     8 15X,*PRINT FLIGHT SEQUENCE (-1=NO, 0=ALL, N=NUMBER)...*,I5/
     9 15X,*NUMBER  OF SPECTRUM SUMMATIONS TO BE PRINTED.....*,I5/
     B 15X,*PROGRAM TERMINATION FLAG (0=NORMAL, ............*,/
     C 15X,*.........................M=STOP AFTER M FLIGHTS)*,I5/
     D 15X,*ALTERNATE FLIGHT SEQUENCE FLAG....................*,I5/
     E 15X,*.......(0=RANDOM, 1=LO-HI, 2=HI-LO, 3=LO-HI-LO)..*,/
     G 15X,* NUMBER OF HIGHEST PEAKS PER FLIGHT TO BE PRINTED*,I5)
      WRITE (6,1002)  KF,KC
```

```
1002 FORMAT (
    1 15X,*STARTING VALUE FOR THE GENERATION OF RANDOM......*,/
    2 15X,*...FLIGHT NUMBERS (0 = DEFAULT DEFINED AS 11111).*,I5/
    3 15X,*STARTING VALUE FOR THE RANDOM CYCLE GENERATION...*,I5/
    4 15X,*..................(0 = DEFAULT DEFINED AS 12345).*,)
      NRAN = IABS(IRAN)
      NPI = IABS(IPI)
      IVP = IABS(KVP)
      READ (5,*) CLIP,CLIV,FACTOR,ELIMP
      WRITE(6,2) CLIP,CLIV,FACTOR,ELIMP
    2 FORMAT (
    X 15X,*PEAK CLIPPING VALUE IS.....................*,F10.0/
    1 15X,*VALLEY CLIPPING VALUE IS...................*,F10.0/
    2 15X,*MULTIPLICATION FACTOR IS...................*,F10.5/
    3 15X,*CYCLE ELIMINATION PEAK VALUE IS............*,F10.0//)
      LEFT=NSIZE-2*NFT
      IF(LEFT.LT.0) CALL ERROR(1,LEFT,NSIZE,NFT)
      CALL INPUTF(NFT,N(1),N(NFT+1),N(2*NFT+1),NST,IRAN,IPI,KVP )
      RETURN
      END
```

```
      SUBROUTINE INPUTF (NFT,NF,NS,A,NST,IRAN,IPI,KVP )
C*********  TFIS SUBROUTINE READS IN THE NUMBER OF FLIGHTS    ***********
C*********  AND THE NUMBER OF SEGMENTS IN EACH FLIGHT TYPE.   ***********
C*********  IT ALSC SETS UP SOME IMPLIED EQUIVALENCES TC      ***********
C*********  THE A ARRAY.                                      ***********
C*********  SUBROUTINES CALLEC - ERROR, INF1F2                ***********
      DIMENSION NF(NFT),NS(NFT),A(1)
      CCMMCN NPG,TITLE(20),NSIZE,LEFT,IERR,IRS,IUIL,NPI,NRAN,
     1      IVP,IFI,NXY,CLIP,CLIV,FACTOR,ELIMP,ELIMV,IPFS,NPSS,IPTF,
     2      IAFS,NEXT,NOW,IFRS
C*********  READ IN THE NUMBER OF FLIGHTS IN EACH FLIGHT TYPE  *********
      READ(5,*) NF
C*********  READ IN THE NUMBER OF SEGMENTS IN EACH FLIGHT TYPE  ********
      REAC (5,*) NS
    1 FCRMAT(1X,15I7)
C*********  COMPUTE THE TOTAL NUMBER OF FLIGHTS    *********************
C*********        ANC THE TOTAL NUMBER OF SEGMENTS.  *********************
      NST=0
      NTF=0
      DO 2  I=1,NFT
      NST=NST+NS(I)
      NTF=NTF+NF(I)
    2 CCNTINUE
C*********  CALCULATE THE STARTING POINT FOR EACH ARRAY WITHIN *********
C*********  THE A ARRAY.                                        *********
      MPI=1
      MRAN=MPI+NPI
      MVP=MRAN+NRAN
      MXY=MVP+IVP
      MISS=MXY+2*NXY
      MF1=MISS+NPSS
      MF2=MF1+NST
      MFRS =MF2+NST
      MN = MFRS + (2*IFRS)
      LEFT=LEFT-(2*(NST+NXY+IFRS)+IVP+NRAN+NPI+NPSS)
      IF(LEFT.LT.0) CALL ERROR(1,LEFT,NSIZE,NST)
      CALL INF1F2(NST,NFT,NS,A(MF1),A(MF2),A(MN),NF1ST,NF2ST,NTF,NF,
     1           A(MPI),A(MRAN),A(MVP),A(MXY),IRAN,IPI,KVP,A(MISS),
     2           A(MFRS) )
      RETURN
      ENC
```

```
      SUBROUTINE INF1F2(NST,NFT,NS,IF1,IF2,N,NF1ST,NF2ST,NTF,NF,
     1                  PI,RAN,VP,XY,IRAN,IPI,KVP,ISS,NFRS )
C*********  THIS SUBROUTINE READS AND PRINTS THE REMAINDER OF THE  *****
C*********  INPUT DATA.  IT SETS UP THE CORE STORAGE REQUIRED FOR  *****
C*********  THE CALLS TO SUBROUTINES INMMN, GENFL, AND GENAFS.     *****
C*********  SUBROUTINES CALLED - ERROR, GENAFS, GENFL, INMMN,      *****
C*********                       NEWPG, OPENMS, WTAPE              *****
      DIMENSION NS(NFT),IF1(NST),IF2(NST),N(1),NF(NFT),NFRS(2,1)
      DIMENSION PI(1),RAN(1),VP(1),XY(2,1),ISS(1)
      COMMON NPG,TITLE(20),NSIZE,LEFT,IERR,IRS,IUIL,NPI,NRAN,
     1       IVP,IFI,NXY,CLIP,CLIV,FACTOR,ELIMP,ELIMV,IPFS,NPSS,IPTF,
     2       IAFS,NEXT,NOW,IFRS,MAXFP
      LINE = 60
 1003 FORMAT (1H ,*INF1F2*,13I7)
 1005 FORMAT (1H ,*INF1F2*,10F12.2)
      IF( IPI.GT.0 )  GO TO 40
C*********  READ IN AND PRINT ALL PEAK LEVELS (IPI ) 0)  *************
      READ(5,*) (PI(I),I=1,NPI)
      GO TO 60
C*********  READ IN AND PRINT FIRST AND LAST PEAK LEVELS.  *************
C*********  LET THE PROGRAM COMPUTE EVENLY SPACED VALUES.  *************
   40 READ(5,*)  PI(1),PI(NPI)
      NC = NPI - 1
      DEL = (PI(NPI) - PI(1)) / NC
      DO  45  I=2,NC
   45 PI(I) = PI(I-1) + DEL
   60 WRITE(6,11) (PI(I),I=1,NPI)
   11 FORMAT(5X,*INPUT PEAK LEVELS FOR SPECTRUM SUMMATION*/(10F12.0))
      IF( IRAN.GT.0 )  GO TO 30
C*********  READ IN AND PRINT ALL RANGE LEVELS (IRAN ) 0)  *************
      READ(5,*) (RAN(I),I=1,NRAN)
      GO TO 50
C*********  READ IN AND PRINT FIRST AND LAST RANGE LEVELS.  ************
C*********  LET THE PROGRAM COMPUTE EVENLY SPACED VALUES.   ************
   30 READ(5,*)  RAN(1),RAN(NRAN)
      ND = NRAN - 1
      DEL = (RAN(NRAN)-RAN(1)) / ND
      DO  35  I=2,NC
   35 RAN(I) = RAN(I-1) + DEL
   50 WRITE(6,12) (RAN(I),I=1,NRAN)
   12 FORMAT(5X,*INPUT RANGE LEVELS FOR SPECTRUM SUMMATION*/(10F12.0))
      IF( KVP.GT.0 )  GO TO 70
C*********  READ IN AND PRINT ALL THE VALLEY/PEAK RATIOS (KVP ) 0) *****
      READ(5,*) (VP(I),I=1,IVP)
      GO TO 80
C*********  READ IN AND PRINT THE FIRST AND LAST VALLEY/PEAK RATIOS. ***
C*********  LET THE PROGRAM COMPUTE EVENLY SPACED VALUES.            ***
   70 READ(5,*)  VP(1),VP(IVP)
      NC = IVP - 1
```

```
      DEL = (VP(IVP) - VP(1)) / NC
      DC  75  I=2,NC
   75 VP(I) = VP(I-1) + DEL
   80 WRITE(6,13) (VP(I),I=1,IVP)
   13 FORMAT(5X,*INPUT VALLEY/PEAK RATICS FCR SPECTRUM SUMMATICN*/
     1       (5X,18F7.3))
      IF( NXY.EQ.0 )  GC TO 90
C*********  READ IN AND PRINT THE VALLEY/PEAK RATIC VS RANGE CURVE  ****
      READ(5,*) ((XY(I,J),I=1,2),J=1,NXY)
      WRITE(6,14) ((XY(I,J),I=1,2),J=1,NXY)
   14 FORMAT(5X,
     1 *INPUT VALLEY/PEAK RATIO VS RANGE CURVE FCR RANGE TRUNCATION*/
     2 8X,5(F7.3,F10.0,4X)/8X,5(F7.3,F10.0,4X))
   90 IF(NPSS.EQ.0 )  GO TO 95
C*********  READ IN AND PRINT THE NUMBERS OF THE FLIGHTS   ************
C*********  AFTER WHICH A SPECTRUM SUMMATION IS TO BE PRINTED  *********
      READ(5,*)  (ISS(I),I=1,NPSS)
      WRITE(6,15)  (ISS(I),I=1,NPSS)
   15 FORMAT(5X,*INPUT FLIGHT NUMBERS FOR SPECTRUM SUMMATICN PRINT*/
     1 (5X,20I6))
   95 REWINC 3
C*********  READ IN AND PRINT A6PA REFERENCE RUN, CASE NUMBER,  ********
C*********  AND SEGMENTS FRCM TAPE UNIT 3.                      ********
      IF( LINE.LT.55 )  GO TO 20
      CALL  NEW PG
      LINE = 4
   20 IS2 = 0
      JSS = 0
      NST = 0
      WRITE (6,1004)
 1004 FCRMAT (1H0,4X,*FLIGHT*,6X,*RR*,5X,*CASE*,5X,*A6PA SEGMENTS*/)
      LINE = LINE + 2
      DO  200  I=1,NFT
      IS1 = IS2 + 1
      NST = NST + NS(I)
      IF( NST.EQ.JSS )  GO TO 240
      IF( NST.LT.JSS )  GO TO 210
  230 READ (3)  IRR,ICASE,ISEG
      ISP = JSS
      JSS = JSS + ISEG
      IF( NST.LT.JSS )  GO TO 210
  240 IS2 = ISEG
      WRITE (6,1002)  I,IRR,ICASE,IS1,IS2
      LINE = LINE + 1
      IF( NST.NE.JSS )  GO TO 230
      IS2 = 0
      GC TO 200
  210 IS2 = NST - ISP
      WRITE (6,1002)  I,IRR,ICASE,IS1,IS2
      LINE = LINE + 1
 1002 FORMAT (1H ,I8,4X,I6,2X,I6,6X,2I6)
  200 CONTINUE
      REWINC 3
```

```
      NF1ST=0
      NF2ST=0
      IS1=1
      IS2=0
C*********  READ IN AND PRINT F1 AND F2 SEGMENTS  ***********************
      IF( LINE.LT.50 )  GO TO 10
      CALL  NEW PG
      LINE = 4
   10 DO 1  I=1,NFT
      IS2=IS2+NS(I)
      READ(5,*) (IF1(J),J=IS1,IS2)
      READ(5,*) (IF2(J),J=IS1,IS2)
      WRITE(6,5) I,NF(I),NS(I)
    5 FORMAT(1H0,4X,*FLIGHT TYPE*,I5,* HAS*,I6,* FLIGHTS AND*,I5,
     1* A6PA SEGMENTS*)
      WRITE(6,6)(IF1(J),J=IS1,IS2)
    6 FORMAT(8X,*F1 SEGMENTS*,35I3)
      WRITE(6,7)(IF2(J),J=IS1,IS2)
    7 FORMAT(8X,*F2 SEGMENTS*,35I3)
      MAX=0
      MBX=0
      DO 2  J=IS1,IS2
      IF(IF1(J).GT.MAX) MAX=IF1(J)
      IF(IF2(J).GT.MBX) MBX=IF2(J)
    2 CONTINUE
      NF1ST=NF1ST+MAX
      NF2ST=NF2ST+MBX
      LINE = LINE + 4
      IF( LINE.LT.51 )  GO TO 1
      LINE = 4
      CALL  NEW PG
    1 IS1=IS1+NS(I)
  134 FORMAT(1X,5F12.2)
    3 FORMAT(1X,*IN F1  F2*,20I6)
      IF( IFRS.EQ.0 )  GO TO 85
C*********  READ IN USER SPECIFIED FLIGHT SEQUENCE  *********************
      READ (5,*)  ((NFRS(I,J),I=1,2),J=1,IFRS)
      WRITE (6,1001)  ((NFRS(I,J),I=1,2),J=1,IFRS)
 1001  FORMAT (1H0,4X,*INPUT FLIGHT SEQUENCE*/(1X,8(I4,I6,5X)))
C*********  CALCULATE THE STARTING POINT WITHIN THE  N  ARRAY   *********
C*********  (WHICH IS REALLY THE  A  ARRAY) FOR EACH ARRAY      *********
C*********  USED IN SUBROUTINE INMMN.                           *********
   85 M3 = (LEFT - NST) / 3
      MAX = 1
      MIN = MAX + M3
      MCY = MIN + M3
      MS2 = MCY + M3
      CALL IN MMN (NFT,NS,IF1,IF2,NST,N(MAX),N(MIN),N(MCY),
     1            NF1ST,NF2ST,MMN,M3,NS1,NS2,NTF,NF,N(MS2) )
      DO 4  I=1,MMN
      N(MMN+I)=N(MIN+I-1)
    4 N(2*MMN+I)= N(MCY+I-1)
C*********  CALCULATE THE STARTING POINT WITHIN THE  N  ARRAY   *********
```

```
C*********   (WHICH IS REALLY THE  A  ARRAY) FOR EACH ARRAY      *********
C*********   USED IN SUBROUTINE GENFL.                           *********
      MHPEAK=(2+IRS)*MMN+1
      MIDP = MHPEAK + MAXHP
      MPMAX = MIDP + MAXHP
      MCY = MPMAX
      MINDEX = MCY
      MJTN = MINDEX
      MFF = MJTN
      IF( IAFS.EQ.0 )  GO TO 99
      MCY = MPMAX + NTF
      MINDEX = MCY + NTF
      MJTN = MINDEX + (NTF + 1)
      MFF = MJTN + NTF
      CALL  OPEN MS  ( 4,N(MINDEX),(NTF+1),0 )
   99 MRR = MFF + 2*NFT
      NNRAN = NRAN + 1
      NNPI = NPI + 1
      MPR=MRR+( IVP+2)* NNRAN
      MIRR=MPR+( IVP+2)* NNPI
      NRMAX = MAX0(NNRAN,NNPI)
      MMS = MIRR + (IVP+2)*NRMAX
      JLEFT = LEFT
      LEFT = LEFT - MMS
      IF(LEFT.LT.0) CALL ERROR(1,LEFT,NSIZE,MMS)
      MAXSS=LEFT/2
      IF( (IFI.NE.0).AND.(IAFS.NE.0) )  REWIND IFI
C*********   GENERATE THE FLIGHT SEQUENCE  ********************************
      CALL  GEN FL(NF,NFT,NTF,NS,NST,N(1),N(MMN+1),N(2*MMN+1),MMN,IF2,
     1             N(MMS),MAXSS,PI,RAN,VP,XY,N(MRR),NNRAN,N(MPR),NNPI,
     2             N(MIRR),NRMAX,ISS,N(MPMAX),N(MCY),N(MFF),NFRS,
     3             N(MHPEAK),N(MIDP),N(MJTN) )
C*********   CALCULATE THE STARTING POINT WITHIN THE  N  ARRAY  *********
C*********   (WHICH IS REALLY THE  A  ARRAY) FOR EACH ARRAY     *********
C*********   USED IN SUBROUTINE GENAFS.                         *********
      MREC = MIRR + (IVP+2)*NRMAX
      MAFS = MREC + NTF
      MFLG = MAFS + NTF
      MMS = MFLG + NTF
      LEFT = JLEFT
      LEFT = LEFT - MMS
      IF(LEFT.LT.0) CALL ERROR(1,LEFT,NSIZE,MMS)
      IF( IAFS.EQ.0 )  RETURN
C*********   GENERATE THE ALTERNATE FLIGHT SEQUENCE  *******************
      CALL  GEN AFS  ( N(MCY),N(MPMAX),N(MREC),N(MAFS),N(MFLG),NTF,
     1                 N(MMS),N(MRR),NNRAN,N(MPR),NNPI,N(MIRR),NRMAX,PI,
     2                 RAN,VP,N(MFF),ISS,NFT,N(MHPEAK),N(MIDP),N(MJTN) )
      IF( IFI.EQ.0 )  RETURN
C*********   SAVE THE ALTERNATE FLIGHT SEQUENCE ON MAGNETIC TAPE  *******
      MMS = MFLG
      CALL  W TAPE  ( N(MREC),N(MCY),N(MMS),NTF,NTF )
      RETURN
      END
```

```
      SUBROUTINE NEW PG
C*********  THIS SUBROUTINE PRINTS CUT A HEADING AT THE TCP OF   ********
C*********  EACH NEW PAGE.  THE HEADING INCLUDES CCNSECUTIVE      ********
C*********  PAGE NUMBERING AND THE USER INPUT TITLE.               ********
C*********  SUBROUTINES CALLED - NONE                             ********
      COMMON NPG,TITLE(20)
      WRITE(6,1) NPG,TITLE
    1 FORMAT(*1     SPECTRUM LOADING SEQUENCE GENERATICN PRCGRAM*,
     1      15X,*PAGE*,I5//5X,*JOB TITLE  *,20A4//)
      NPG=NPG+1
      RETURN
      END
```

```
      SUBROUTINE IN MMN (NFT,NS,IF1,IF2,NST,SMAX,SMIN,NCY,NF1ST,
     1                   NF2ST,MMN,M3,NS1,NS2,NTF,NF,IS2 )
C*********  THIS SUBROUTINE SETS UP THE CORE STORAGE    *****************
C*********  REQUIRED FOR THE CALLS TO SUBROUTINE AMMN   *****************
C*********  SUBROUTINES CALLED -  AMMN, ERROR           *****************
      DIMENSION NS(NFT),IF1(NST),IF2(NST),SMAX(M3),SMIN(M3),NCY(M3)
      DIMENSION NF(NFT),IS2(NST)
    3 FORMAT(1X,*IN MMN*,15I7)
      JF1=1
      JF2=1
      JS1=1
      JS2=1
      MMN=1
C*********  GET THE MAXIMUM STRESSES, MINIMUM STRESSES, AND NUMBER   ****
C*********  OF CYCLES FOR EACH FLIGHT TYPE.  ALSO DO COMBINING OF    ****
C*********  SEGMENTS AND CYCLES.                                     ****
      DO 1 I=1,NFT
      NSS=NS(I)
      CALL AMMN (NSS,IF1(JF1),IF2(JF2),IF1(JS1),IS2(JS2),NS1,NS2,
     1           SMAX(MMN),SMIN(MMN),NCY(MMN),M3-MMN,NNM,NF(I),
     2           NCY(MMN),I  )
      JF1=JF1+NSS
      JF2=JF2+NSS
      JS1=JS1+NS1
      JS2=JS2+NS2
      NS(I)=JS2 - 1 +(NST+1)*MMN
      MMN=MMN+NNM
      IF( MMN.GE.M3 )  CALL  ERROR ( 3,MMN,M3,I )
    1 CONTINUE
      MMN=MMN-1
      JS2 = JS2 - 1
    4 FORMAT (1H ,*IN MMN*,10F12.2)
C*********  AFTER PERFORMING THE COMBINING OF  SEGMENTS,  **************
C*********  RECREATE THE F2 ARRAY.                        **************
      DO  200  I=1,JS2
  200 IF2(I) = IS2(I)
      RETURN
      END
```

```
      SUBROUTINE AMMN (NSS,IF1,IF2,IS1,IS2,NS1,NS2,SMAX,SMIN,NCY,
     1  MM,MMN,NF,BCY,ICFT )
C********** THIS SUBROUTINE PERFORMS THE COMBINING OF SEGMENTS BASED   *
C********** ON THE USER INPUT ARRAYS, F1 AND F2, PRODUCING NEW          *
C********** MAXIMUM STRESS, MINIMUM STRESS AND NUMBER OF CYCLE ARRAYS. *
      COMMON SKIP(25),IUIL
      DIMENSION IF1(NSS),IF2(NSS),IS1(1),IS2(1),SMAX(MM),SMIN(MM),
     1          NCY(MM)
      DIMENSION AMAX(25),AMIN(25),ACY(25),BCY(MM)
  133 FORMAT(1X,*AMMN *,2I6,19I4)
  134 FORMAT(1X,*AMMN *,6F12.2)
      JS1=0
      MMN=0
C********* SUMMATION OF CYCLES ************************************
C********* F1 COMBINING OF SEGMENTS ************************************
      DO 1 I=1,NSS
C********** READ IN NUMBER OF GROUPS, PEAK VALUE, VALLEY ***************
C********** NUMBER OF CYCLES SEQUENCE FROM UTILITY TAPE  ***************
      READ(IUIL) KMMN,(AMAX(K),AMIN(K),ACY(K),K=1,KMMN)
      KF1=IF1(I)
      IF(KF1.EQ.0) GO TO 41
      IF(KF1.LE.JS1) GO TO 2
      JS1=JS1+1
      DO 3 J=1,KMMN
      MMN=MMN+1
      SMAX(MMN)=AMAX(J)
      SMIN(MMN)=AMIN(J)
       BCY(MMN)= ACY(J)
    3 CONTINUE
   11 IS1(JS1)=MMN
      GO TO 1
    2 K2=IS1(KF1)
      IF(KF1.EQ.1) K1=1
      IF(KF1.NE.1) K1=IS1(KF1-1)+1
      K=1
      DO 4 J=K1,K2
      BCY(J)=BCY(J)+ACY(K)
    4 K=K+1
   41 IF2(I)=0
    1 CONTINUE
      NS1=JS1
      K1=1
      L=1
C********* ROUND OFF CYCLES TO NEAREST INTEGER AND ELIMINATE CYCLES
C********* THAT ARE LESS THAN 0.5
      DO 31 I=1,NS1
      K2=IS1(I)
      DO 32 J=K1,K2
      NCY(L)=BCY(J)+ .5000001
      SMAX(L) = SMAX(J)
      SMIN(L) = SMIN(J)
```

```
      IF(NCY(L).GT.0) L=L+1
   32 CCNTINUE
      K1=K2+1
      IF( L.LE.1 )  CALL   ERROR  ( 7,I,NSS,K2 )
   31 IS1(I)=L-1
      MMN=L-1
C********** MOVE DOWN TO THE END CF THE ARRAYS SC THAT F2 **************
C********** COMBINING OF SEGMENTS CAN BE DONE               **************
      K=MMN
      J=MM
      DC 5 I=1,MMN
      SMAX(J)=SMAX(K)
      SMIN(J)=SMIN(K)
       NCY(J)= NCY(K)
      J=J-1
    5 K=K-1
      NCEL=J-K
      KS1=1
C********* F2 CCMBINING OF SEGMENTS *************************************
      NSS1=NSS+1
      DC  51  I=1,NSS
      IS2(I) = 0
      KKQ = K
      KQ = KKQ + 1
      KF1 = 0
      DO 7 J=1,NSS1
      IF(J.EQ.NSS1) GO TO 28
      IF( IF2(J).NE.0 )  KF1 = KF1 + 1
      IF(IF2(J).NE.I) GO TO 7
      GC TO 6
   28 IF( KKQ.EQ.K )  GO TO 52
      KQQ=KQ+IS2(I)-1
      KK=0
      DC 22 L=KQ,KQQ
   22 KK=KK+NCY(L)
      AA=(KK-1)/NF
      IA=AA
      NCY(K+1)=(IA+1)*NF-KK
      IF(NCY(K+1).EQ.0) GO TO 21
C********* FORMULATE FICTITIOUS LCAD LEVEL *************************
      K=K+1
      IS2(I)=IS2(I)+1
      MAX=0
      MIN=0
      KQ1=KQ+1
      IF(KQ1.GT.KQQ) GO TO 26
      IF(SMAX(KQ1).NE.SMAX(KQ)) MAX=1
      IF(SMIN(KQ1).NE.SMIN(KQ)) MIN=1
       IF(MIN.NE.0) GO TO 24
   26 SMIN(KQQ+1)=SMIN(KQ)
      SMAX(KQQ+1)=SMIN(KQ)
      GC TO 21
   24 IF(MAX.NE.0) GO TC 25
```

```
      SMAX(KQQ+1)=SMAX(KQ)
      SMIN(KQQ+1)=SMAX(KQ)
      GC TO 21
   25 SMAX(KQQ+1)=(SMAX(KQ)+SMIN(KQ))/2.C
      SMIN(KQQ+1)=SMAX(KQQ+1)
   21 CCNTINUE
C***  WRITE(2,133) I,KQ,KQQ,KQ1,K,MAX,MIN,IA,IS2(I-1),(NCY(L),L=KQ,K)
C***  WRITE(2,134) (SMAX(L),SMIN(L),L=KQ,K)
      GC TO 7
    6 K2=IS1(KF1)+NCEL
      IF(KF1.EQ.1) K1=1+NCEL
      IF(KF1.NE.1) K1=IS1(KF1-1)+1+NDEL
      DC 9 L=K1,K2
      K=K+1
      SMAX(K)=SMAX(L)
      SMIN(K)=SMIN(L)
      BCY(K) = BCY(L)
    9 NCY(K) = NCY(L)
      IS2(I)=IS2(I)+K2-K1+1
    7 CONTINUE
   51  CONTINUE
      I = NSS1
   52 NS2=I-1
      MMN=K
      DO 10 I=2,NS2
      IS2(I)=IS2(I)+IS2(I-1)
   10 CCNTINUE
      LINE = 60
      IS = 1
      IAA = 0
      IA = 0
      BAA = 0.0
      AAA = 0.0
C********** PRINT OUT AFTER SEGMENT COMBINING *************************
      DC  2C  I=1,MMN
      AT = NCY(I)
      AA = AT / NF
      AAA = AAA + AA
      IA = IA + NCY(I)
      IF( LINE.LT.49 )  GC TC 30
      CALL  NEW PG
      WRITE (6,1001)  ICFT
 1001 FCRMAT (1HU,4X,*FLIGHT TYPE*,I4,7X,*F2*,9X,*MIN.*,7X,*MAX.*,/,
     *          25X,*SECMENT*,5X,*STRESS*,5X,*STRESS*,6X,*CYCLES*,
     1          6X,*CYCLES/FLIGHT*,/)
      LINE = 0
   30 WRITE (6,1002)  IS,SMIN(I),SMAX(I),NCY(I),AA
      LINE = LINE + 1
      IF( IS2(IS).GT.I )  GC TO 2C
      IAA = IA - IAA
      BAA = AAA - BAA
      WRITE (6,1003)  IAA,BAA
      LINE = LINE + 2
```

```
      IS = IS + 1
      IAA = IA
      BAA = AAA
 1002 FORMAT (1H ,24X,I4,F14.0,F11.0,I12,5X,F12.4)
   20 CCNTINUE
      WRITE (6,1003)  IA,AAA
 1003 FCRMAT (1H ,53X,I12,5X,F12.4,/)
      RETURN
      ENC
```

CALL OPENMS (u,ix,lngth,t)†

Opens mass storage file and informs Record Manager that this file is word addressable. If an existing file is called, the master index is read into the area specified by the program. u is the unit designator. ix is the first word address of the index in central memory. lngth is the length of the index buffer; for a name index, lngth ≥ 2 * (number of records in file) + 1; for a number index, lngth ≥ number of records in file + 1. t = 1 file is referenced through a name index; t = 0 file is referenced through a number index.

Example:

```
PROGRAM MS1 (TAPE3)
DIMENSION INDEX (11), DATA (25)
CALL OPENMS (3,INDEX,11,0)
```

```
      SUBROUTINE GEN FL(NF,NFT,NTF,NS,NST,SMAX,SMIN,NCY,NMM,IS2,SMM,
     1                  MAXSS,PI,RAN,VP,XY,RR,NNRAN,PR,NNPI,IRR,NRMAX,ISS,
     2                  PMAX,MCY,NFF,IPF,HPEAK,IDPEAK,JTN )
C*********   THIS SUBROUTINE GENERATES THE SEQUENCE CF FLIGHTS.    ********
C*********   DEPENDING UPON INPUT, THE SEQUENCE CAN BE RANDOMLY     ********
C*********   GENERATED OR USER SPECIFIED..                          ********
C*********   SUBROUTINES CALLED - DISTRD, GENCY, PRNTSS             ********
      COMMON NPG,TITLE(20),NSIZE,LEFT,IERR,IRS,IUIL,NPI,NRAN,
     1       IVP,IFI,NXY,CLIP,CLIV,FACTOR,ELIMP,ELIMV,IPFS,IPSS,IPTF,
     2       IAFS,NEXT,NOW,IFRS,MAXHP,IKF,IKC
      INTEGER  RR(NNRAN,1),PR(NNPI,1)
      DIMENSION NF(NFT),NS(NFT),SMAX(NMM),SMIN(NMM),IS2(NST),XY(2,1)
      DIMENSION SMM(2,MAXSS),NCY(NMM,1),PI(1),RAN(1),VP(1)
      DIMENSION PMAX(NTF),MCY(NTF),JTN(NTF),IRR(NRMAX,1),IJJ(6),ISS(1)
      DIMENSION  IPF(2,1),NFF(2,NFT),HPEAK(1),IDPEAK(1)
  133 FORMAT(1X,*GEN FL*,2CI6)
  134 FORMAT(1X,*GEN FL*,10F12.2)
      KF=11111
      IF( IKF.NE.0 )  KF = IKF
      KC=12345
      IF( IKC.NE.0 )  KC = IKC
      NPRNT = IPFS
      IF( IPFS.EQ.0 ) NPRNT = NTF
      NPSS = ISS(1)
      IF( IPSS.EQ.0 )  NPSS = NTF
      NPTF = IPTF
      IF( IPTF.EQ.0 )  NPTF = NTF
      IS = 1
      KLINE = 60
      NFT2 = 2 * NFT
C*********  ZERO OUT THE TABLES USED IN THE SPECTRUM SUMMATION  ********
      DO  86  I=1,NFT2
   86 NFF(I,1) = 0
      DO  85  I=1,6
   85 IJJ(I) = 0
      IF( IRS.EQ.1 )  GO TO 2
      DO  3  I=1,NMM
    3 NCY(I,2) = NCY(I,1)
    2 IIVP = IVP + 2
      DO 41 I=1,IIVP
      DO 42 J=1,NNRAN
   42 RR(J,I)=0
      DO 43 J=1,NNPI
   43 PR(J,I)=0
   41 CONTINUE
      IF( MAXHP.EQ.0 )  GO TO 50
C*********  ZERO OUT HIGHEST PEAK ARRAYS  ******************************
      DO  55  J=1,MAXHP
      HPEAK(J) = 0.0
```

```
      IDPEAK(J) = 0
   55 CONTINUE
   50 SMM(1,1)=-1.E20
      SMM(2,1)=1.E20
      NST1 = NST + 1
      I1 = 2
      I2 = 1
      JPF = 1
      DC 1 II=1,NTF
      IF( IFRS.EQ.0 )  GO TO 81
C*********  THE SEQUENCE OF FLIGHT NUMBERS IS SPECIFIED BY INPUT  ******
      JF = IPF(1,JPF)
      IPF(2,JPF) = IPF(2,JPF) - 1
      IF( IPF(2,JPF).EQ.0 )  JPF = JPF + 1
      GO TO 82
C*********  THE SEQUENCE OF FLIGHT NUMBERS IS RANDOMLY GENERATED  ******
   81 CALL DIST RD(NF,NFT,KF,JF)
   82 NFF(1,JF) = NFF(1,JF) + 1
      MMN=NS(JF)/NST1
      IF(JF.NE.1) JS2=NS(JF-1)-NST1*(NS(JF-1)/NST1)+1
      IF(JF.EQ.1)JS2=1
      NSS=NS(JF)-MMN*NST1-JS2+1
C***  WRITE(2,133) II,NMM,IRS
C***  WRITE(2,133) NSS,JF,MMN,NST1,JS2,JPF,NF,NS,NFT,NTF,MAXSS
      IF( II.EQ.NTF )  I2 = 0
C*********  GENERATE CYCLE SEQUENCE  ************************************
      CALL GEN CY(IS2(JS2),SMAX(MMN),SMIN(MMN),NCY(MMN,1),NF(JF),
     1            SMM(1,1),MAXSS,NSS,NCY(MMN,2),PI,RAN,VP,XY,RR,NNRAN,
     2            PR,NNPI,KC,I1,I2,IJJ,II,JF,NPRNT,KLINE,AMAX,KCY,NPTF)
      IF( IAFS.EQ.0 )  GO TO 40
C*********  SAVE THE HIGHEST PEAK AND NUMBER OF CYCLES FOR EACH FLIGHT
C*********  TO BE USED FOR THE ALTERNATE FLIGHT SEQUENCE
      PMAX(II) = AMAX
      MCY(II) = 2 * KCY
      JTN(II) = JF
      GO TO 65
   40 IF( MAXHP.EQ.0 )  GO TO 65
C*********  DETERMINE AND SAVE THE HIGHEST PEAK AND CORRESPONDING  *****
C*********  FLIGHT NUMBER FOR MAXHP (SPECIFIED BY INPUT) NUMBER    *****
C*********  OF FLIGHTS                                             *****
      HP = AMAX
      III = II
      DO  20  I=1,MAXHP
      IF( HP.LE.HPEAK(I) )  GO TO 20
      DO  30  K=I,MAXHP
      TMAX = HPEAK(K)
      HPEAK(K) = HP
      HP = TMAX
      IJF = IDPEAK(K)
      IDPEAK(K) = III
      III = IJF
   30 CONTINUE
   20 CONTINUE
```

```
   65  I1 = 1                                                             00000980
       NFF(2,JF) = NFF(2,JF) + KCY                                        00000990
       NF(JF)=NF(JF)-1                                                    00001000
       IF( (II.NE.NPSS).CR.(IAFS.NE.C) )  GO TC 10                        00001010
C***   WRITE (2,133) IJJ                                                  00001020
C***   WRITE (2,133) MCY                                                  00001030
C***   WRITE (2,134) PMAX                                                 00001040
C*********  IF NO ALTERNATE FLIGHT SEQUENCE IS DESIRED THEN  ***********00001042
C*********  PRINT THE SPECTRUM SUMMATICN TABLES                ***********00001044
       CALL  PRNT SS  ( RR,NNRAN,PR,NNPI,IRR,NRMAX,IIVP,RAN,VP,PI,IS,     00001050
     1                  NFF,ISS,II,NTF,NFT,HPEAK,IDPEAK )                 00001060
   10  IF( II.EQ.NPTF )  GO TO 11                                         00001070
    1  CONTINUE                                                           00001080
   11  RETURN                                                             00001090
       ENC                                                                00001100
```

```
      SUBROUTINE  GEN AFS  ( MCY,PMAX,LREC,AFS,IFLAG,NTF,SMM,RR,NNRAN,
     1                       PR,NNPI,IRR,NRMAX,PI,RAN,VP,NFF,ISS,NFT,
     2                       HPEAK,IDPEAK,JTN )
      INTEGER    RR(NNRAN,1),PR(NNPI,1),IRR(NRMAX,1)
C*********  THIS SUBROUTINE RECRDERS THE RANDOM FLIGHT SEQUENCE  *******
C*********  INTO LO-HI, HI-LO, LO-HI-LO FLIGHT SEQUENCE BASED    *******
C*********  UPON THE MAXIMUM STRESS PER FLIGHT.  IT PERFORMS A   *******
C*********  TYPE 1 EDIT BEFORE DOING A SPECTRUM SUMMATION.       *******
C********* SUBROUTINES CALLED - NEWPG, PRNTSS,REED, REDIT1, SPSUM  *****
      COMMON   ISKIP(28),IVP,ISKP(7),IPFS,IPSS,IPTF,IAFS,SPA(3),MAXHP
      DIMENSION  MCY(NTF),PMAX(NTF),LREC(NTF),AFS(NTF),IFLAG(NTF),N1(3)
      DIMENSION   IADD(3),FORM(3,3),TITLE(5),SMM(2,1),IJJ(6),JTN(NTF)
      DIMENSION   PI(1),RAN(1),VP(1),NFF(1),ISS(1),HPEAK(1),IDPEAK(1)
      DATA FORM /4H    ,4H(HI ,4HLO) ,4H    ,4H(LO ,4HHI) ,
     4           4H (LO,4H HI ,4HLO) /
      DATA TITLE /4HFLT.,4H NO.,4H  MA,4HX. P,4HEAK /
      NPRNT = IPFS
      IF( IPFS.EQ.0 )  NPRNT = NTF
      EPS = 1.0
      N1(1) = NTF + 1
      N1(2) = 0
      N1(3) = (NTF/2) + 1
      IADD(1) = -1
      IADD(2) =  1
      IADD(3) = -1
      INC = IADD(IAFS)
      IS = N1(IAFS)
      NPRNT = IPFS
      IF( IPFS.EQ.0 ) NPRNT = NTF
      NPSS = ISS(1)
      IF( NPSS.EQ.0 )  NPSS = NTF
      NPTF = IPTF
      IF( IPTF.EQ.0 )  NPTF = NTF
      IIVP = IVP + 2
      DO 41 I=1,IIVP
      DO 42 J=1,NNRAN
   42 RR(J,I)=0
      DO 43 J=1,NNPI
   43 PR(J,I)=0
   41 CONTINUE
      DO  400  I=1,NTF
  400 IFLAG(I) = 0
C*********  REORDER THE FLIGHT SEQUENCE INTO LO-HI, HI-LO, OR LO-HI-LO *
C*********  SEQUENCE DEPENDING UPON THE IAFS FLAG SPECIFIED BY INPUT    *
      DO  600  II=1,NTF
      PHI = -1.E20
```

```
      DO   500   I=1,NTF
      IF( PMAX(I).LT.PHI )   GO TO 5CO
      IF( IFLAG(I).GT.0 )   GC TC 5CC
      PHI = PMAX(I)
      IFL = I
  500 CCNTINUE
      IF( IAFS.NE.3 )   GO TO 300
      IADD(3) = -1 * IACD(3)
      INC = IACD(3) * (II - 1)
  300 IS = IS + INC
      AFS(IS) = PHI
      LREC(IS) = IFL
      IFLAG(IFL) = 1
C*********   FIND THE HIGHEST STRESSES AND ASSOCIATED FLIGHT NUMBER   ****
      IF( MAXHP.EQ.0 )   GO TO 60C
      HP = PHI
      III = IS
      DC   20   JJ=1,MAXHP
      IF( HP.LE.HPEAK(JJ) )   GO TO 2C
      DO   30   K=JJ,MAXHP
      TMAX = HPEAK(K)
      HPEAK(K) = HP
      HP = TMAX
      JF = IDPEAK(K)
      IDPEAK(K) = III
      III = JF
   30 CCNTINUE
   20 CCNTINUE
  600 CONTINUE
C*********   PRINT THE NEW ORDER OF FLIGHTS AND THE LARGEST STRESS   *****
C*********   PER FLIGHT                                               *****
      LINE = 50
      DO   200   I=1,NTF,5
      N4 = I
      N2 = I + 4
      IF( N2.GT.NTF )   N2 = NTF
      N3 = N2 - N4 + 1
      LINE = LINE + 1
      IF( LINE.LT.45 )   GO TO 250
      CALL   NEW PG
      WRITE (6,1001)   (FORM(J,IAFS),J=1,3)
 1001 FCRMAT (1H ,15X,3A4,*CRDER CF FLIGHTS (LARGEST PEAK / FLIGHT)*,//)
      LINE = 0
      WRITE (6,1003)   ((TITLE(K),K=1,5),J=1,N3)
 1003 FORMAT (1H ,5(5A4,4X))
  250 WRITE (6,1002)   (LREC(J),AFS(J),J=N4,N2)
 1002 FORMAT (1H ,5(I6,F13.2,5X))
  200 CCNTINUE
      IS = 1
      KLINE = 60
      DC   700   I=1,NTF
      NCY = MCY(LREC(I))
      CALL   REED   ( SMM,LREC(I),NCY )
```

```
C********* REPEAT AN EDIT 1 BETWEEN FLIGHTS BECAUSE OF REORDERING ******
      NCY = NCY / 2
      CALL  R EDIT 1  ( SMM,IJJ,NCY,EPS )
C********* SPECTRUM SUMMATION ******************************************
      CALL  SP SUM  ( SMM,1,NCY,PMAX(I),VP,RAN,PI,RR,NNRAN,PR,NNPI )
   88 FORMAT (1H ,*GEN AFS*,4I5,4F10.C)
      IF( I.GT.NPRNT )  GO TO 9CC
      IF( KLINE.LT.53 )  GC TC 65C
      CALL  NEW PG
      KLINE = 5
  650 WRITE (6,1004)  I,LREC(I),JTN(LREC(I)),NCY
      KLINE = KLINE + 2
C********  PRINT THE NEW FLIGHT NUMBER, THE OLD FLIGHT NUMBER,  ********
C********* FLIGHT TYPE NUMBER, NUMBER OF CYCLES AND THE         ********
C********* SEQUENCE OF MAXIMUM AND MINIMUM STRESSES.            *******
      DO  800  JJ=1,NCY,5
      J1 = JJ
      J2 = J1 + 4
      IF( J2.GT.NCY )  J2 = NCY
      IF( KLINE.LT.55 )  GC TC 75C
      CALL  NEW PG
      KLINE = 5
 1004 FORMAT (1H0,* NEW FLIGHT NUMBER*,I5,* OLD FLIGHT NUMBER*,I5,
     1         *   IS TYPE NUMBER*,I3,*   NUMBER OF CYCLES*,I7,
     2         *  SEQUENCE FOLLOWS *)
  750 WRITE (6,1005)  ((SMM(K,J),K=1,2),J=J1,J2)
 1005 FORMAT (1H ,10F12.2)
      KLINE = KLINE + 1
  800 CONTINUE
  900 IF( I.NE.NPSS )  GO TO 72C
C*********  PRINT THE SPECTRUM SUMMATION  ******************************
C*********  PRINT THE SPECTRUM SUMMATION                        ********
      CALL  PRNT SS  ( RR,NNRAN,PR,NNPI,IRR,NRMAX,IIVP,RAN,VP,PI,IS,
     1                 NFF,ISS,I,NTF,NFT,HPEAK,IDPEAK )
  720 IF( I.EQ.NPTF )  GO TO 73C
  700 CONTINUE
  730 RETURN
      END
```

```
      SUBROUTINE GEN CY( IS2,SMAX,SMIN,NCY,NF,SMM,MAXSS,NSS,KCY,          00000030
     1                   PI,RAN,VP,XY,RR,NNRAN,PR,NNPI,KC,I1,I2,IJJ,II,   00000040
     2                   JF,NPRNT,KLINE,PMAX,MCY,INTF )                    00000050
C*********   THIS SUBROUTINE GENERATES THE CYCLE SEQUENCE FOR THE     *****00000051
C*********   GIVEN FLIGHT.  IT THEN PERFORMS ALL EDITING OF THE       *****00000052
C*********   CYCLES.  IF A SAVE TAPE IS SPECIFIED AND NO ALTERNATE    *****00000053
C*********   FLIGHT SEQUENCE IS DESIRED, THEN THE CYCLE SEQUENCE      *****00000054
C*********   IS WRITTEN DIRECTLY ONTO AN OUTPUT TAPE.  OTHERWISE      *****00000055
C*********   THE CYCLE SEQUENCE IS WRITTEN ONTO TEMPORARY MASS        *****00000056
C*********   STORAGE.                                                 *****00000057
C*********   SUBROUTINES CALLED - DISTRD, ERROR, NEWPG, REDIT1,       *****00000058
C*********                        RITE, SPSUM, WTAPE                       00000059
      COMMON NPG,TITLE(20),NSIZE,LEFT,IERR,IRS,IUIL,NPI,NRAN,              00000060
     1       IVP,IFI,NXY,CLIP,CLIV,FACTOR,ELIMP,ELIMV,IPFS,IPSS,IPTF,      00000070
     2       IAFS                                                          00000080
      DIMENSION IS2(NSS),SMAX(1),SMIN(1),NCY(1),SMM(2,MAXSS)               00000090
      DIMENSION KCY(1),IJJ(1)                                              00000100
      INTEGER  RR,PR                                                       00000110
      DIMENSION PI(1),RAN(1),VP(1),XY(2,1),RR(NNRAN,1)                     00000120
      DIMENSION PR(NNPI,1)                                                 00000130
  133 FORMAT(1X,*GEN CY*,I3,19I4)                                          00000140
  134 FORMAT(1X,*GEN CY*,5F12.2)                                           00000150
 1001 FORMAT (1H0,*  FLIGHT NUMBER*,I4,*  IS TYPE NUMBER*,I3,              00000160
     1 *    NUMBER OF CYCLES*,I7,*    SEQUENCE FOLLOWS *)                  00000170
      IMM=1                                                                00000180
      KMM=1                                                                00000190
      K1=1                                                                 00000200
      K0=0                                                                 00000210
      IF( II.EQ.1 )  NERR5 = 0                                             00000212
      EPS = 1.                                                             00000220
      PMAX = -1.E20                                                        00000230
      DO 1 I=1,NSS                                                         00000240
      K2=IS2(I)                                                            00000250
      NCS=0                                                                00000260
      K3=K2-K0                                                             00000270
      LMM = IMM                                                            00000280
      DO 2 J=K1,K2                                                         00000290
    2 NCS=NCS+NCY(J)                                                       00000300
      LCY=NCS/NF                                                           00000310
      DO 11 J=1,LCY                                                        00000320
C*******   IF PAIRED VALLEY PEAK COUPLING (IRS=1) IS SPECIFIED,    ********00000323
C*******   FIND THE RANDOM CYCLE.  IF INDIVIDUAL VALLEY PEAK       ********00000324
C*******   COUPLING (IRS=2) IS SPECIFIED, FIND THE RANDOM PEAK.    ********00000325
      CALL DIST RD(NCY(K1),K3,KC,IC)                                       00000330
      IMM=IMM+1                                                            00000340
      IF( IMM.LT.MAXSS )  GO TO 50                                         00000350
      CALL  ERROR  ( 4,IMM,NSIZE,JF )                                      00000360
      STOP 7002                                                            00000370
   50 SMM(1,IMM)=SMIN(IC+KO)                                               00000380
       NCY(IC+KO)=NCY(IC+KO)-1                                             00000390
```

```
      IF(IRS.EQ.2) GO TO 5
      SMM(2,IMM)=SMAX(IC+KO)
      GO TO 3
C*********  IF INDIVIDUAL VALLEY PEAK COUPLING (IRS=2) IS     *************
C*********  SPECIFIED, FIND THE RANDOM VALLEY.                *************
    5 CALL DIST RD(KCY(K1),K2,KC,JC)
      SMM(2,IMM)=SMAX(JC+KO)
      KCY(JC+KO)=KCY(JC+KO)-1
    3 CONTINUE
C********* EDIT 1 ***************************************************
C********* CHECK TO SEE IF MINIMUM IS REALLY A VALLEY AND NOT AN  ******
C********* INTERMEDIATE POINT ON THE WAY TO A PEAK                ******
      IF( SMM(1,IMM-1).NE.SMM(2,IMM-1) )  GO TO 42
      IF( SMM(1,IMM-1).GT.SMM(1,IMM) )  SMM(1,IMM-1) = SMM(1,IMM)
      SMM(2,IMM-1) = SMM(2,IMM)
      IMM = IMM - 1
      IJJ(2) = IJJ(2) + 1
      GO TO 11
   42 IF(SMM(1,IMM).LT.SMM(2,IMM-1)) GO TO 6
      IJJ(1)=IJJ(1)+1
   88 FORMAT(1X,4I5,4F10.0)
      SMM(2,IMM-1)=SMM(2,IMM)
      IMM=IMM-1
    6 IF( ABS(SMM(1,IMM)-SMM(2,IMM)).LT.EPS )  GO TO 8
      IF( SMM(1,IMM)-SMM(2,IMM) )  11,8,9
C********* DROP OUT FICTITIOUS LOAD LEVELS ***************************
    8 SMM(1,IMM) = SMM(2,IMM)
      IF( SMM(2,IMM).LT.SMM(2,IMM-1) )  GO TO 11
      SMM(2,IMM-1) = SMM(2,IMM)
   41 IMM = IMM - 1
      IJJ(3)=IJJ(3)+1
      GO TO 11
C********* VALLEY GREATER THAN PEAK IN THIS SEGMENT *******************
C********* INTERCHANGE VALLEY AND PEAK **********************************
    9 TEMP = SMM(1,IMM)
      SMM(1,IMM) = SMM(2,IMM)
      SMM(2,IMM) = TEMP
   11 CONTINUE
C***  WRITE (2,133)  (IJJ(J),J=1,3)
      KO=K2
      K1=KO+1
    1 CONTINUE
C********* EDIT 2 ***************************************************
      KMM = IMM
      IMM = I1 - 1
      DO  12  JMM = I1,KMM
      IMM = IMM + 1
      SMM(1,IMM) = SMM(1,JMM)
      SMM(2,IMM) = SMM(2,JMM)
C*********  ELIMINATE RANGES BELOW AN INPUT RANGE VERSUS R CURVE  ******
      RANGE=SMM(2,IMM)-SMM(1,IMM)
      R=SMM(1,IMM)/SMM(2,IMM)
C***  WRITE (2,133)  JMM,IMM,I1,KMM
```

```
C***  WRITE (2,134)  SMM(1,IMM),SMM(2,IMM),RANGE,R                      00000900
      IF( NXY.LE.1 )  GC TO 23                                          00000910
      IF(XY(1,1).LE.R)  GC TO 15                                        00000920
      IF( NERR5.LT.6 )  CALL  ERRCR  ( 5,II,IMM,1 )                     00000930
      NERR5 = NERR5 + 1                                                 00000932
      K = 2                                                             00000940
      GC TO 22                                                          00000950
   15  DO 21 K=2,NXY                                                    00000960
      IF(XY(1,K).GE.R) GO TO 22                                         00000970
   21 CONTINUE                                                          00000980
      IF( NERR5.LT.6 )  CALL ERROR (5,II,IMM,2 )                        00000990
      NERR5 = NERR5 + 1                                                 00000992
      K=NXY                                                             00001000
   22 RQ=XY(2,K)+(R-XY(1,K-1))*(XY(2,K)-XY(2,K-1))/(XY(1,K)-XY(1,K-1))  00001010
      IF(RANGE.GT.RQ) GC TO 23                                          00001020
      IF( IMM.EQ.1 )  GC TO 3C                                          00001030
      IF( SMM(2,IMM).LE.SMM(2,IMM-1) )  GO TO 30                        00001040
      SMM(2,IMM-1) = SMM(2,IMM)                                         00001050
      GC TO 25                                                          00001060
   30 IF( (JMM+1).GT.KMM )  GC TO 25                                    00001070
      IF( (SMM(1,IMM).GE.SMM(1,JMM+1)) )  GC TO 25                      00001080
      SMM(1,JMM+1) = SMM(1,IMM)                                         00001090
   25 IJJ(4)=IJJ(4)+1                                                   00001100
   24 IMM=IMM-1                                                         00001110
      IJJ(6) = IJJ(6) + 1                                               00001120
      GC TC 12                                                          00001130
C*********  ELIMINATE PEAKS BY INPUT VALUES.         ***************00001132
C*********  CLIP PEAKS ANC/CR VALLEYS BY INPUT VALLES.  ***************00001134
   23 IF(SMM(2,IMM).GT.ELIMP) GO TO 24                                  00001140
      IF(SMM(2,IMM).LT.CLIP) GO TO 26                                   00001150
      IF(SMM(1,IMM).LT.CLIP)  GO TO 14                                  00001160
C********* IF BOTH VALLEY AND PEAK ABOVE PEAK CLIPPING ELIMINATE CYCLE  00001170
C****** OR IF BOTH VALLEY AND PEAK BELOW VALLEY CLIPPING ELIMINATE CYCLE00001180
   16 IMM=IMM-1                                                         00001190
      IJJ(5) = IJJ(5) + 1                                               00001200
      GC TO 12                                                          00001210
   14 SMM(2,IMM)=CLIP                                                   00001220
  26  IF(SMM(1,IMM).GT.CLIV) GO TC 12                                   00001230
      IF(SMM(2,IMM).LT.CLIV)  GO TC 16                                  00001240
      SMM(1,IMM) = CLIV                                                 00001250
   12 CCNTINUE                                                          00001260
      MCY = IMM - 1                                                     00001270
      IF( MCY.EQ.0 )  GC TO 75                                          00001280
C********* REPEAT A TYPE 1 EDIT  *********************************      00001290
      CALL  R EDIT 1  ( SMM,IJJ,IMM,EPS )                               00001300
      MCY = IMM - 1                                                     00001310
      IF( MCY.EQ.0 )  GC TO 75                                          00001320
C********** MULTIPLICATION FACTOR ***********************************00001330
      DC  40  I=1,IMM                                                   00001340
      DC  40  J=1,2                                                     00001350
   40 SMM(J,I) = SMM(J,I) * FACTOR                                      00001360
   13 I3 = IMM - I2                                                     00001370
C********* SPECTRUM SUMMATION ***************************************00001380
```

```
      CALL   SP SUM  ( SMM,I1,I2,PMAX,VP,RAN,PI,RR,NNRAN,PR,NNPI )
      IF( IAFS.EQ.0 )  GO TO 66
C*********8  WRITE CYCLE SEQUENCE ONTO TEMPORARY MASS STORAGE  **********
      CALL   RITE  ( SMM(1,2),II,(2*MCY) )
      GO TO 62
C********* IF NO ALTERNATE FLIGHT SEQUENCE IS DESIRED, BUT A SAVE  *****
C********* TAPE IS SPECIFIED, THEN WRITE THE CYCLE SEQUENCE        *****
C********* DIRECTLY ONTO THE OUTPUT TAPE.                           *****
   66 IF( IFI.NE.0 )  CALL  W TAPE  ( II,(IMM-I2),SMM,INTF,I1 )
      IF( II.GT.NPRNT )  GO TO 62
      IF( KLINE.LT.53 )  GO TO 65
      CALL  NEW PG
      KLINE = 5
   65 WRITE (6,1001)  II,JF,MCY
      KLINE = KLINE + 2
      DO  60  JJ=2,IMM,5
      J1 = JJ
      J2 = J1 + 4
      IF( J2.GT.IMM )  J2 = IMM
      IF( KLINE.LT.55 )  GO TO 70
      CALL  NEW PG
      KLINE = 5
   70 WRITE (6,1134)  ((SMM(I,J),I=1,2),J=J1,J2)
      KLINE = KLINE + 1
   60 CONTINUE
 1134 FORMAT (1H ,10F12.2)
C*********  ELIMINATE THE MULTIPLICATION FACTOR WHEN CARRYING THE  *****
C*********  LAST CYCLE OVER INTO THE NEXT FLIGHT.                  *****
   62 SMM(1,1) = SMM(1,IMM) / FACTOR
      SMM(2,1) = SMM(2,IMM) / FACTOR
      RETURN
   75 WRITE (6,1001)  II,JF,MCY
      CALL  ERROR  ( 6,II,JF,MCY )
      RETURN
      END
```

```
      SUBROUTINE DIST RD (N,M,K,L)
C*********   THIS SUBROUTINE USES A RANDOM NUMBER GENERATOR  ************
C*********   TO SELECT A FLIGHT NUMBER OR A CYCLE DEPENDING  ************
C*********   UPON THE CONTENTS OF THE INPUT ARRAY.  THE      ************
C*********   SEQUENCE OF FLIGHTS OR CYCLES IS ESTABLISHED BY ************
C*********   BY SELECTING WITHOUT REPLACING A CHOSEN NUMBER. ************
C*********   SUBROUTINES CALLED -  RANIC                      ***********
      DIMENSION N(M)
    3 FORMAT (1X,I3,19I4)
      IF(M.EQ.0) RETURN
      NS=0
      DO 1  I=1,M
      NS=NS+N(I)
    1 CONTINUE
      CALL  RANIC ( K, R )
      LR=NS*R+1
      MS=0
      DO 2  I=1,M
      MS=MS+N(I)
      L=I
      IF(MS.GE.LR) GO TO 5
    2 CONTINUE
    5 CONTINUE
    4 FORMAT(1X,I12,F12.6,12I4)
      RETURN
      END
```

```
      SUBROUTINE  RANIC  ( K, R )
C*********    THIS SUBROUTINE GENERATES A PSEUDC RANDOM NUMBER, WHICH
C*********    LIES BETWEEN 0 AND 1 INCLUSIVE, SUCCESSIVE ENTRIES WILL
C*********    YIELD A SERIES OF NUMBERS WHICH CONFORM TC A UNIFORM
C*********    DISTRIBUTICN.  THE SERIES REPEATS AFTER APPROXIMATELY
C*********    10**6 NUMBERS.
C
C             K = I/O = GENERATING INTEGER ARGUMENT.  K MUST BE
C                       INITIALIZED TO ANY NON-ZERO VALUE.  THEREAFTER
C                       K IS MODIFIED BY THE SUBROUTINE AND SHOULD NOT
C                       BE CHANGED BY THE USER.
C             R =  O  = THE GENERATED RANDOM NUMBER
      DATA IMAX /2147483647/
      K=K*2051
      IF(K.LT.0) K=K+IMAX+1
      K=MOD(K, 4194304 )
      R=FLOAT(K)/ 4194304.
      RETURN
      END
```

```
      SUBROUTINE  REEC  ( A,MAT,NSIZE )
C*********  THIS SUBROUTINE READS AND ENTIRE SEQUENCE CF     **************
C*********  MAXIMUM AND MINIMUM STRESSES FOR CNE FLIGHT      **************
C*********  FROM TEMPCRARY MASS STORAGE.                     **************
C*********  SUBROUTINES CALLED - READMS                      **************
      DIMENSION A(NSIZE)
      CALL  READ MS  ( 4,A,NSIZE,MAT )
      RETURN
      END
```

CALL READMS (u,fwa,n,k)†

Transmits data from mass storage to central memory. fwa is the central memory address of the first word of the record. n is the number of central memory words transferred. Number index $k = 1 \leq k \leq$ lngth - 1. Name index k = any 60-bit quantity except ± 0. u is the unit designator.

Example:

```
CALL READMS(3,DATA,25,6)
```

```
      SUBROUTINE  R EDIT 1  ( SMM,IJJ,IMM,EPS )
C*********  THIS SUBROUTINE PERFORMS A REPEAT OF A TYPE 1 EDIT  ********
C*********  SUBROUTINES CALLED - NONE                            ********
      DIMENSION  SMM(2,1),IJJ(1)
      KMM = IMM
      IMM =  1
      DO  70  JMM=2,KMM
      IMM = IMM + 1
      SMM(1,IMM) = SMM(1,JMM)
      SMM(2,IMM) = SMM(2,JMM)
C********* CHECK TO SEE IF MINIMUM IS REALLY A VALLEY AND NOT AN
C********* INTERMEDIATE POINT ON THE WAY TO A PEAK
      IF( SMM(1,IMM-1).NE.SMM(2,IMM-1) )  GO TO 72
      IF( SMM(1,IMM-1).GT.SMM(1,IMM) )  SMM(1,IMM-1) = SMM(1,IMM)
      SMM(2,IMM-1) = SMM(2,IMM)
      IMM = IMM - 1
      IJJ(2) = IJJ(2) + 1
      GO TO 70
   72 IF(SMM(1,IMM).LT.SMM(2,IMM-1)) GO TO 76
      IJJ(1)=IJJ(1)+1
      SMM(2,IMM-1)=SMM(2,IMM)
      IMM=IMM-1
   76 IF( ABS(SMM(1,IMM)-SMM(2,IMM)).LT.EPS )  GO TO 73
      IF( SMM(1,IMM)-SMM(2,IMM) )  70,73,79
C********* CROP OUT FICTITIOUS LOAD LEVELS ****************************
   73 SMM(1,IMM) = SMM(2,IMM)
      IF( SMM(2,IMM).LT.SMM(2,IMM-1) )  GO TO 70
      SMM(2,IMM-1) = SMM(2,IMM)
   71 IMM = IMM - 1
      IJJ(3)=IJJ(3)+1
      GO TO 70
C********* VALLEY GREATER THAN PEAK IN THIS SEGMENT *******************
C********* INTERCHANGE VALLEY AND PEAK ********************************
   79 TEMP = SMM(1,IMM)
      SMM(1,IMM) = SMM(2,IMM)
      SMM(2,IMM) = TEMP
   70 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE  SP SUM   ( SMM,I1,I3,PMAX,VP,RAN,PI,RR,NNRAN,PR,NNPI ) 00000020
C*********  THIS SUBROUTINE GENERATES THE TABLES REQUIRED FOR    ********00000030
C*********  THE SPECTRUM SUMMATICN PRINT OUT CF RANGE VS.        ********00000031
C*********  VALLEY/PEAK RATIO AND PEAK VS. VALLEY/PEAK RATIO.    ********00000032
C*********  SUBROUTINES CALLEC - NCNE                            ********00000034
      COMMON  ISKIP(26),NPI,NRAN,IVP                                    00000040
      DIMENSION  SMM(2,1),RAN(1),VP(1),PI(1)                            00000050
      INTEGER   RR(NNRAN,1),PR(NNPI,1)                                  00000060
      DC 51 KMM=I1,I3                                                   00000070
      IF( SMM(2,KMM).GT.PMAX )  PMAX = SMM(2,KMM)                       00000080
      RANGE=SMM(2,KMM)-SMM(1,KMM)                                       00000090
      R=SMM(1,KMM)/SMM(2,KMM)                                           00000100
C*********  TEST RATIO (MIN./MAX.) AGAINST INPUT VALLEY/PEAK RATIO  ****00000108
   27 DC 28 K=1,IVP                                                     00000110
      IF(VP(K).GE.R) GO TO 29                                           00000120
   28 CONTINUE                                                          00000130
      K = IVP + 1                                                       00000140
C*********  TEST RANGE AGAINST INPUT RANGE INTERVALS  ******************00000148
   29 DO 30 L=1,NRAN                                                    00000150
      IF(RAN(L).GE.RANGE) GC TO 31                                      00000160
   30 CCNTINUE                                                          00000170
      L = NRAN + 1                                                      00000180
   31 RR(L,K)=RR(L,K)+1                                                 00000190
C*********  TEST MAXIMUM STRESS AGAINST INPUT PEAK INTERVALS  **********00000198
      DO 32 M=1,NPI                                                     00000200
      IF(PI(M).GE.SMM(2,KMM)) GC TC 33                                  00000210
   32 CCNTINUE                                                          00000220
      M = NPI + 1                                                       00000230
   33 PR(M,K)=PR(M,K)+1                                                 00000240
   51 CCNTINUE                                                          00000250
      RETURN                                                            00000260
      END                                                               00000270
```

```
      SUBROUTINE  PRNT SS  ( RR,NNRAN,PR,NNPI,IRR,NRMAX,IIVP,RAN,VP,PI,
     1                 IS,NFF,ISS,II,NTF,NFT,PMAX,MFT )
C*********  THIS SUBROUTINE PRINTS OUT THE TWO SPECTRUM SUMMATION    *****
C*********  TABLES, RANGE VERSUS VALLEY/PEAK RATIO AND PEAK VERSUS   *****
C*********  VALLEY/PEAK RATIO.  IT ALSO PRINTS OUT THE HIGHEST       *****
C*********  PEAKS AND THEIR CORRESPONDING FLIGHT NUMBER.             *****
C*********  SUBROUTINES CALLED - NEWPG                               *****
      COMMON  ISKIP(26),NPI,NRAN,IVP,SKIP(8),IPSS,SSKIP(5),MAXHP
      INTEGER  RR(NNRAN,1),PR(NNPI,1)
      DIMENSION  RAN(1),VP(1),PI(1),IRR(NRMAX,1)
      DIMENSION  NFF(2,1),TITLE1(6),TITLE2(6),TITLE3(2),TITLE4(4)
      DIMENSION  FORM(4),FMAT(6),ISS(1),PMAX(1),MFT(1)
   40 FORMAT(30X,*SPECTRUM SUMMATION FOR A TOTAL OF*,I5,* FLIGHTS AND*,
     1        I7,* CYCLES*)
   44 FORMAT(1H0,2X,*VALLEY/PEAK RATIO *,6X,6F16.2)
   45 FORMAT(14X,*RANGE *)
   47 FORMAT(9X,2F7.0,7(2I7,2X))
 1047 FORMAT (1X,*BELOW OR EQUAL *,F7.0,7(2I7,2X))
 1048 FORMAT (9X,* ABOVE *,F7.0,7(2I7,2X))
   48 FORMAT(15X,*PEAK *)
      DATA  TITLE1 /4HFLIG,4HHT   ,4HNUMB,4HER   ,4H NUM,4HBER /
      DATA  TITLE2 /4H TYP,4HE    ,4HFLIG,4HHTS  ,4H CYC,4HLES /
      DATA  TITLE3 /4HTOTA,4HL    /
      DATA  TITLE4 /4HPEAK,4H    F,4HLIGH,4HT    /
      DATA  FORM /4H(1H+,4H     ,4H,2A4,4H)     /
      DATA  FMAT /4H,T30,4H,T46,4H,T62,4H,T78,4H,T94,4HT110/
C*********  PRINT THE RANGE VERSUS VALLEY/PEAL RATIC  *****************
      IVP3 = IIVP - 1
      DO  25  I=1,NNRAN
      DO  25  J=1,IVP3
   25 RR(I,IIVP) = RR(I,IIVP) + RR(I,J)
      DO  20  J=1,IIVP
   20 IRR(NNRAN,J) = RR(1,J)
      DO  24  I=1,NRAN
      DO  24  J=1,IIVP
   24 IRR(NNRAN-I,J) =  IRR(NNRAN-I+1,J) + RR(I+1,J)
      DO  80  I=1,NRAN
      DO  80  J=1,IIVP
   80 IRR(I+1,J) = IRR(I,J) - RR(I,J)
      LINE = 50
      IVP1 = 1
   50 IVP2 = IVP1 + 5
      IF( IVP2.GT.IIVP )  IVP2 = IIVP
      IVP3 = IVP2
      IF( IVP2.GT.IVP )  IVP3 = IVP
      IF( LINE.LT.45 )  GO TO 55
      LINE = 0
      CALL NEW PG
      WRITE (6,40)  II,IRR(1,IIVP)
   55 WRITE (6,44)  (VP(J),J=IVP1,IVP3)
      IF( IVP2.NE.IIVP )  GO TO 51
```

```
      ITAB = IVP2 -IVP1 + 1
      FORM(2) = FMAT(ITAB)
      WRITE (6,FORM)  TITLE3
   51 WRITE(6,45)
      WRITE (6,1047)  RAN(1),(RR(1,J),IRR(1,J),J=IVP1,IVP2)
      LINE = LINE + 4
      DO 46 I=2,NRAN
      WRITE(6,47) RAN(I-1),RAN(I),(RR(I,J),IRR(I,J),J=IVP1,IVP2)
      LINE = LINE + 1
      IF( (LINE.LT.48).OR.(I.EQ.NRAN) )  GO TO 46
      LINE = 0
      CALL  NEW PG
      WRITE (6,40)  II,IRR(1,IIVP)
      WRITE (6,44)  (VP(J),J=IVP1,IVP3)
      IF( IVP2.NE.IIVP )  GO TO 52
      WRITE (6,FORM)  TITLE3
   52 WRITE (6,45)
   46 CONTINUE
      WRITE (6,1048)  RAN(NRAN),(RR(NNRAN,J),IRR(NNRAN,J),J=IVP1,IVP2)
C*********  PRINT THE PEAK VERSUS VALLEY/PEQK RATIO  ******************
      IVP1 = IVP2 + 1
      IF( IVP1.LE.IIVP )  GO TO 50
      IVP3 = IIVP - 1
      DO  35  I=1,NNPI
      DO  35  J=1,IVP3
   35 PR(I,IIVP) = PR(I,IIVP) + PR(I,J)
      DO  30  J=1,IIVP
   30 IRR(NNPI,J) = PR(1,J)
      DO  34  I=1,NPI
      DO  34  J=1,IIVP
   34 IRR(NNPI-I,J) = IRR(NNPI-I+1,J) + PR(I+1,J)
      DO  90  I=1,NPI
      DO  90  J=1,IIVP
   90 IRR(I+1,J) = IRR(I,J) - PR(I,J)
      IF( LINE.LT.35 )  GO TO 60
      LINE = 0
      CALL  NEW PG
      WRITE (6,40)  II,IRR(1,IIVP)
   60 IVP1 = 1
   70 IVP2 = IVP1 + 5
      IF( IVP2.GT.IIVP ) IVP2 = IIVP
      IVP3 = IVP2
      IF( IVP2.GT.IVP )  IVP3 = IVP
      IF( LINE.LT.45 )  GO TO 75
      LINE = 0
      CALL  NEW PG
      WRITE (6,40)  II,IRR(1,IIVP)
   75 WRITE (6,44)  (VP(J),J=IVP1,IVP3)
      IF( IVP2.NE.IIVP )  GO TO 71
      ITAB = IVP2 -IVP1 + 1
      FORM(2) = FMAT(ITAB)
      WRITE (6,FORM)  TITLE3
   71 WRITE(6,48)
```

```
      WRITE (6,1047)  PI(1),(PR(1,J),IRR(1,J),J=IVP1,IVP2)
      LINE = LINE + 4
      DO 49 I=2,NPI
      WRITE(6,47) PI(I-1),PI(I),(PR(I,J),IRR(I,J),J=IVP1,IVP2)
      LINE = LINE + 1
      IF( (LINE.LT.48).OR.(I.EQ.NPI) )  GO TO 49
      LINE = 0
      CALL  NEW PG
      WRITE (6,40)   II,IRR(1,IIVP)
      WRITE (6,44)   (VP(J),J=IVP1,IVP3)
      IF( IVP2.NE.IIVP )  GO TO 72
      WRITE (6,FORM)  TITLE3
   72 WRITE (6,48)
   49 CONTINUE
      WRITE (6,1048)  PI(NPI),(PR(NNPI,J),IRR(NNPI,J),J=IVP1,IVP2)
      IVP1 = IVP2 + 1
      IF( IVP1.LE.IIVP )  GO TO 70
      JLINE = 50
      DO  77 J=1,NFT,4
      M1 = J
      M2 = J + 3
      IF( M2.GT.NFT )  M2 = NFT
      M3 = M2 - M1 + 1
      IF( JLINE.LT.45 )  GO TO 76
      CALL NEW PG
      WRITE (6,40)   II,IRR(1,IIVP)
      WRITE (6,1003)  ((TITLE1(K),K=1,6),KK=1,M3)
      WRITE (6,1003)  ((TITLE2(K),K=1,6),KK=1,M3)
 1003 FORMAT (1H ,(4(6A4,5X)))
      JLINE = 0
   76 WRITE (6,1001)  (K,(NFF(I,K),I=1,2),K=M1,M2)
 1001 FORMAT (1H ,4(I5,2I9,6X))
      JLINE = JLINE + 1
   77 CONTINUE
C*********  PRINT THE HIGHEST PEAKS AND THEIR CORRESPONDING  **********
C*********  FLIGHT NUMBER                                     **********
   65 IF( MAXHP.EQ.0 )  GO TO 12
      M3 = 6
      IF( M3.GT.MAXHP )  M3 = MAXHP
      IF( JLINE.LT.45 )  GO TO 68
      CALL  NEW PG
      JLINE = 0
   68 WRITE (6,1004)  MAXHP
 1004 FORMAT (1H0,//,10X,*THE*,I4,* HIGHEST PEAKS*)
      WRITE (6,1005)  ((TITLE4(K),K=1,4),KK=1,M3)
 1005 FORMAT (1H ,4X,6(4A4,5X))
      JLINE = JLINE + 3
      DO  67  J=1,MAXHP,6
      M1 = J
      M2 = J + 5
      IF( M2.GT.MAXHP )  M2 = MAXHP
      WRITE (6,1006)  ((PMAX(K),MFT(K)),K=M1,M2)
 1006 FORMAT (1H ,6(F8.2,I8,5X))
```

```
      JLINE = JLINE + 1
   67 CONTINUE
   12 IF( IPSS.EQ.IS )  GO TO 10
      IS = IS + 1
      NPSS = ISS(IS)
      KLINE = 60
      DO 92 J=1,NNRAN
   92 RR(J,IIVP)=0
      DO 93 J=1,NNPI
   93 PR(J,IIVP)=0
   10 RETURN
      END
```

```
      SUBROUTINE ERROR (I,J,K,L)
C*********  THIS SUBROUTINE PRINTS CUT THE ERROR MESSAGES  *************
C*********  SUBROUTINES CALLED - NONE                      *************
      COMMON NPG,TITLE(20),NSIZE,LEFT,IERR,IRS,IUIL,NPI,NRAN,
     1       IVP,IFI,NXY,CLIP,CLIV,FACTOR,ELIMP,ELIMV
      IERR=IERR+1
      WRITE(6,1) IERR,I,J,K,L
    1 FORMAT(* ERROR NUMBER*,I3,* IS TYPE*,I4,* AND INVCLVES*,3I7)
      IF(IERR.GT.900) STOP  701
      RETURN
      END
```

```
      SUBROUTINE  RITE  ( A,MAT,NSIZE )
C*********  THIS SUBROUTINE WRITES AN ENTIRE SEQUENCE  *****************
C*********  OF MAXIMUM AND MINIMUM STRESSES FOR ONE    ****************
C*********  FLIGHT ONTO TEMPORARY STORAGE.             ****************
C*********  SUBROUTINES CALLED - WRITMS                *****************
      DIMENSION A(NSIZE)
      CALL  WRIT MS  ( 4,A,NSIZE,MAT,-1 )
      RETURN
      END
```

CALL WRITMS(u,fwa,n,k,r,s)†

Transmits data from central memory to mass storage. u,fwa,n,k are the same as for READMS. r = +1 rewrites in place. Unconditional request; fatal error is printed if new record length exceeds old record length. r = -1 rewrites in place if space is available, otherwise writes at end of information. r = 0 no rewrite; writes normally at end of information. The r parameter can be omitted if the s parameter is omitted. The default value for r is 0 (normal write).

s = 1 writes subindex marker flag in index control word for this record. s = 0 does not write subindex marker flag in index control word for this record. The s parameter can be omitted; its default value is 0.

The s parameter is included for future random file editing routines. Current routines do not test the flag, but the user should include this parameter in new programs, when appropriate, to facilitate transition to a future edit capability.

Example: `CALL WRITMS (3,DATA,25,NRKEY)`

```
      SUBROUTINE  W TAPE  ( LREC,MCY,SMM,NTF,I1 )                       00000010
C*********  THIS SUBROUTINE WRITES THE FLIGHT NUMBER,   ***************00000011
C*********  THE NUMBER OF CYCLES IN THE FLIGHT, AND THE ***************00000012
C*********  MAXIMUM AND MINIMUM STRESSES FOR THE FLIGHT ***************00000013
C*********  ON THE OUTPUT TAPE.                         ***************00000014
      COMMON  ISKIP(29),IFI,ISK(9),IAFS                                 00000020
      DIMENSION  LREC(NTF),MCY(NTF),SMM(2,1)                            00000030
      IF( IAFS.EQ.0 )   GO TO 300                                       00000040
C*********  WRITE IF ALTERNATE FLIGHT SEQUENCE IS SPECIFIED  ***********00000041
      WRITE (IFI)  NTF                                                  00000050
      DO  200  I=1,NTF                                                  00000060
      MAT = LREC(I)                                                     00000070
      ICY = MCY(LREC(I))                                                00000080
      CALL  REED  ( SMM,MAT,ICY )                                       00000090
      ICY = ICY / 2                                                     00000091
      WRITE (IFI)  I,ICY,((SMM(M,J),M=1,2),J=1,ICY)                     00000100
C***  WRITE (2,1001)  I,ICY,((SMM(M,J),M=1,2),J=1,ICY)                  00000110
 1001 FORMAT (1H ,*W TAPE*,2I5,10X,8F13.2,/(1H ,10F13.2))               00000120
  200 CONTINUE                                                          00000130
C*********  WRITE IF RANDOM ONLY FLIGHT SEQUENCE IS SPECIFIED  *********00000131
      GO TO 400                                                         00000132
  300 IFL = LREC(1)                                                     00000140
      IF( IFL.EQ.1 )  WRITE (IFI) NTF                                   00000150
      ICY = MCY(1)                                                      00000160
      JCY = ICY - I1 + 1                                                00000170
      WRITE (IFI)  IFL,JCY,((SMM(M,J),M=1,2),J=I1,ICY)                  00000180
C***  WRITE (2,1001)  IFL,JCY,((SMM(M,J),M=1,2),J=I1,ICY)               00000181
  400 RETURN                                                            00000190
      END                                                               00000200
```

SECTION VII

OUTPUT

Program A6PD output (excluding A6PA part which is described in Reference 1 and Appendix A) consists of the following printout:

1. Input data. This also includes the relationships between flight type number and program A6PA RR, case and segments.

2. (MIN,MAX,n,n/FLT) arrays for all flight types and F2 segments.

3. Flight-by-flight valley, peak loads sequence for a specified number of flights (IPFS input, card 2).

4. Sepectrum summations

   $$(\text{RANGE, R}) \quad \text{vs} \quad (n,\Sigma n)$$
   $$\text{and} \quad (\text{PEAK, R}) \quad \text{vs} \quad (n,\Sigma n)$$

   The input required to obtain this output is: NPI, NRAN, KVP, NPSS (card 2), PI (card 6), RAN (card 7), VP (card 8) and ISS (card 10). A number of summations, after different number of flights, can be obtained by appropriate NPSS and ISS inputs.

5. Number of flights and cycles for each flight type in the spectrum generated.

6. The magnitude of the N number of largest peaks per flight in the spectrum generated, together with the flight number in which they occur. Input N = MAXHP, card 2.

The complete valley, peak loads sequence may be written out on magnetic tape for further use in fatigue or crack propagation analysis or testing. Input IFI=3 (card 2) if spectrum on tape is desired. The tape contains, first, the total number of flights, and then, flight number, and the number of cycles and the valley, peak loads sequence in that flight, for all the flights.

For sample output printout see pages 208 thru 235 .

SECTION VIII

ERROR CONDITIONS

There are seven special error conditions written for program A6PD. Each error is written out in the following format:

ERROR N IS TYPE M AND INVOLVES J, K, L,

where, N = error message number; the program will stop execution when N exceeds 900 and will print the completion code message STOP 900.

M = error type number.

J,K,L = special description for each error type.

The error conditions and probable steps to be taken to correct the condition, are:

TYPE	DESCRIPTION
1	More core is required for this job. The dimension of the A array in Subroutine RDMAIN and the value of NSIZE which is the dimension of A must be increased. J = amount of additional core needed K = the NSIZE for this run NOTE: Add 2*MAX# of cycles to "J" also.
2	The numbering of the F2 segments is incorrect. J = overall segment number K = number of F2 segments up to this point L = the number of F1 segments up to this point.
3	More core is required for this job. (Message written in Subroutine INMMN.) The dimension of the A array in Subroutine RDMAIN and the value of NSIZE, which is the same as the dimension of A, must be increased. J = the amount of core needed at this point K = NSIZE specified for this run M = flight type number.

4 More core is required for this job. The program is terminated, and a completion code message STOP 9001 is written. (The messages are written in Subroutine GENCY.) The dimension of the A array in Subroutine RDMAIN and the value of NSIZE, which is the same as the dimension of A, must be increased. Look at the indicated flight type.

J = number of cycles allowed for in this run

K = amount of total core specified for this (NSIZE)

L = flight type that exceeds the core requirements

5 The Valley to Peak Ratio of a cycle exceeds the maximum Valley to Peak Ratio input. The cycle is not eliminated. If this is not acceptable, correct the input for the Valley to Peak Ratio versus Range elimination curve. (This error will not be printed beyond five times in one run.)

J = flight number

K = cycle number

L = 1 first end of curve

= 2 second end of curve

6 A flight has been edited until it has zero cycles left. The flight is not written on tape.

J = flight number

K = flight type

L = undefined.

7 In F1 segment (MIN,MAX,n) array cycles (n) for all load levels are zero. Either enter cycles for those A6PA segments making up the F1 segment, or use F1 and F2 flags to eliminate them.

J = F1 segment number

K = total number of segments in the flight type

L = total number of A6PA peaks, valleys and cycles for this segment.

SECTION IX

KEYPUNCH AND DECK SETUP INSTRUCTIONS

Keypunch Instructions

The following instructions apply to A6PD input data only. See Reference 1 for A6PA input data instructions.

1. Use standard keypunch procedures and a standard 80 column data card.

2. All 80 columns of the data card can be used to enter the data.

3. Except in card 1, which is the job description (title) card, see Section V, the data entries must be separated by one or more blanks or a comma.

4. Data cards must be stacked in order by the user, as defined in Section V.

Deck Setup

The program and input data deck setup, where the input data is entered from a punched card deck, is shown in Figure 5. The input data represents information to generate one spectrum loading sequence. An alternate operation procedure, where the input data is read in through CDC utility UPDATE, is given in Appendix B. Job control card content and arrangement varies depending on the particular computer facility and procedures used by the user. Douglas Aircraft Company computer facility and procedures require the following job control cards to run on the CDC computer running under KRONOS:

```
SER, ALØ1.

A6PD, MFL = 155000, T _____, IØ_____, L____.

ACCOUNT,_____, .

CID, Name, etc.

FTN,I=INPUT,L=Ø.

CØPYBR(INPUT,DATA1).

REWIND,DATA1.

REQUEST,TAPE3,NT,LB=KU,D=1600,VSN=RESERVE,T_____.
```

```
LGØ,DATA1.

;;
```

Where,

In Card 2 - MFL = core size required for the program = 155000.

T = estimate of CP time (seconds) to run a job. (Allow approx. 50 seconds per 10^5 cycles of the A6PA spectrum.)

I/O = estimate of input/output processing time (seconds). (This time varies significantly according to the scope of the job. Allow at least 15 seconds per 10^5 cycles of the A6PA spectrum.)

L = estimate of line printout (in thousands). (Printout will depend on the size of the job, the extent of the optional A6PA output and the length of final spectrum sequence printout.)

In Card 7 - T = Number of days that the output tape is to be reserved (saved) in the tape library.

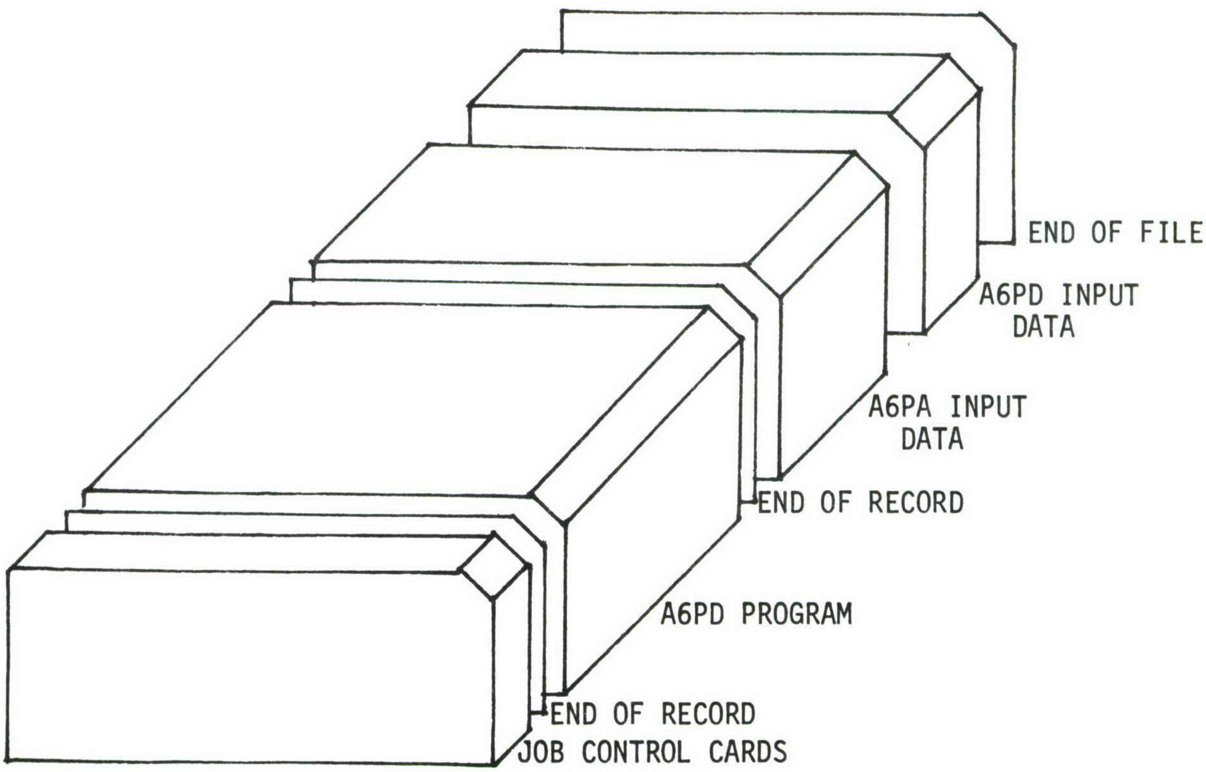


Figure 5. Program A6PD Deck Setup.

SECTION X

TEST CASE

This section presents a test case to check and illustrate the capabilities of program A6PD. The case represents spectrum BS1 from Reference 2.

The spectrum represents two flights of a STOL transport basic and alternate employment mission:

PROGRAM A6PA		FLIGHT	NUMBER OF FLTS. * LDGS.	FLT.HRS. / FLT.	FLIGHT HOURS	A6PD FLIGHT TYPE
RR	CASE					
15	001	1A	1,112	.989	1,099.8	1
15	001	1B	1,416	.989	1,400.4	2
TOTAL:			2,528	.989	2,500.2	

Program A6PA segments and F1 and F2 flags for A6PD segment definition are described in Table 1 . The spectrum represents wing root lower surface stresses. For more background information about these flights, see Reference 2.

The A6PA and A6PD input data is presented on standard load sheets, pages 112 through 165 . The S-N input data (data load sheet V-1.1, page 128) in program A6PA is fictitious so as not to produce any fatigue damage calculation since the objective of this case was only to generate a spectrum loading sequence.

The output is presented on the following pages:

		Pages
A6PA,	- Input data	112-164
	- Output	166-207
A6PD,	- Input data	165
	- Output	208-235

The following output is not shown:

1. A6PA individual segment and summary damage. The damage is zero because of the fictitious S-N data.
2. A6PA, RR 15, Case 2 output. The data is very similar to Case 1, except for differences shown in Table 1 and higher stresses.

Note that, although the spectrum generated represents 2,528 flights, only 50 flights of the loading sequence printout were requested.

Time to run this test case on a CDC computer was:

IO	=	25.3 sec.
CP	=	100.3 sec.
MRU	=	7.8
SRU	=	12.5
Lines printed	=	4,181

TABLE 1

TEST CASE A6PA SEGMENT DESCRIPTION AND F1 AND F2 FLAGS

A6PA SEGMENT	DESCRIPTION	Δg	Flt. 1A Alt. 10^3Ft.	Flt. 1A F1	Flt. 1A F2	Flt. 1B Alt. 10^3Ft.	Flt. 1B F1	Flt. 1B F2
1	Pre-Flt. Taxi	±	SL	1	1	SL	1	1
2	Climb, FD*, Gust	±	0-1	2	2	0-1	2	2
3	Climb, FD, Manv.	±	0-1	2	2	0-1	2	2
4	Climb, FD, Manv.	+	0-1	3	2	0-1	3	2
5	Climb, Gust	±	1-2.5	4	3	1-2.5	4	3
6	Climb, Gust	±	2.5-5	4	3	2.5-5	4	3
7	Climb, Gust	±	5-10	4	3	5-10	4	3
8	Climb, Gust	±	10-20	4	3	10-20	4	3
9	Climb, Gust	±	20-30	4	3	20-30	4	3
10	Climb, Gust	±	30-37.7	4	3	30-35.2	4	3
11	Climb, Manv.	±	1-37.7	4	3	1-35.2	4	3
12	Climb Manv.	+	1-37.7	5	3	1-35.2	5	3
13	Cruise, Gust	±	37.7-38.1	6	4	35.2-35.9	6	4
14	Cruise Manv.	±	37.7-38.1	6	4	35.2-35.9	6	4
15	Cruise, Manv.	+	37.7-38.1	7	4	35.2-35.9	7	4
16	Descent, Gust	±	38.1-30	8	5	35.9-30	8	5
17	Descent, Gust	±	30-20	8	5	30-20	8	5
18	Descent, Gust	±	20-10	8	5	20-10	8	5
19	Descent, Gust	±	10-5	8	5	10-5	8	5
20	Descent, Gust	±	5-3	8	5	5-3	8	5
21	Descent, Gust	±	Not Used	0	0	Not Used	0	0
22	Descent, Manv.	±	38.1-3	8	5	34.1-3	8	5
23	Descent, Manv.	+	38.1-3	9	5	34.1-3	9	5
24	Descent,FD,Gust	±	3-1.5	10	6	3-1.5	10	6
25	Descent,FD,Manv.	±	3-1.5	10	6	3-1.5	10	6
26	Descent,FD,Manv.	+	3-1.5	11	6	3-1.5	11	6
27	Descent,FD,Gust	±	1.5-.5	12	7	1.5-0	12	7
28	Descent,FD,Manv.	±	1.5-.5	12	7	1.5-0	12	7
29	Descent,FD,Manv.	+	1.5-.5	13	7	1.5-0	13	7
30	Descent,FD,Gust	±	.5-0	14	8	Not Used	0	0
31	Descent,FD,Manv.	±	.5-0	14	8	Not Used	0	0
32	Descent,FD,Manv.	+	.5-0	15	8	Not Used	0	0
33	Landing Impact	-	SL	16	9	SL	14	8
34	Post-Flight Taxi	±	SL	17	10	SL	15	9

* FD = Flaps Down

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET I-1.
READ STATEMENT INPUT FORMAT.

Page_____of__________
Prepared by__________
Date______________

KEYPUNCH: INSERT A BLANK CARD FOR EVERY BLANK LINE WHICH PRECEDES OR EXISTS BETWEEN DATA LINES.

```
  1  9

 BASELINE SPECTRUM 1

 RR CASE FLT     FH     FLTS     LANDGS      FH/FLT     FH/LANDG     LANDGS/FLT

 15   1    1A   1099.8  1112     1112         .989       .989             1
 15   2    1B   1400.4  1416     1416         .989       .989             1

    TØTAL:      2500.2  2528     2528         .989       .989             1
```

IREAD = FLAG WHICH SPECIFIES WHETHER WORD DESCRIPTION WILL BE USED. ENTER IN FIRST LINE, CC 3.
= 2, NO.
= 1, YES. IN FIRST LINE, CC 4-6, ENTER FLAG I, THE NUMBER OF LINES (INCLUDING BLANK LINES USED FOR SPACING) REQUIRED TO MAKE THE DESIRED WORD DESCRIPTION. ENTER THE WORD DESCRIPTION ON SUBSEQUENT LINES. IF MORE THAN ONE LOAD SHEET IS REQUIRED TO COMPLETE THE WORD DESCRIPTION, ENTER IREAD AND I VALUES ONLY ON THE FIRST SHEET.

Page____ of______
Prepared by______
Date______

69-70 R R	71-73 CASE	77-80 PROG
15	000	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
L1	1462		1		
IW3	199		1		
IW1	197		2		
IW2	198		1		
IW4	200		2		
IW5	203		1		

L1 = FLAG WHICH SPECIFIES WHETHER THE MULTIPLICATION FACTOR F IN THE CYCLIC LOAD EQUATIONS WILL BE USED.
= 1, YES. (COMPLETE SHEET III-1.2)
= 2, NO. (F = 1.0)

IW3 = FLAG WHICH SPECIFIES WHETHER THE CONSTANT P IN THE CYCLIC LOAD EQUATIONS WILL BE USED.
= 1, YES. (COMPLETE SHEET III-1.3)
= 2, NO. (P = 0)

IW1 = FLAG WHICH SPECIFIES WHETHER GAG CYCLE INFORMATION IS TO BE PRINTED. ENTER ONLY IF I4 ≠ 0.
= 1, YES.
= 2, NO.

IW2 = FLAG WHICH SPECIFIES WHETHER THE SEGMENT SPECTRUM AND DAMAGE INFORMATION IS TO BE PRINTED.
= 1, YES.
= 2, NO.

IW4 = FLAG WHICH SPECIFIES WHETHER SPECTRUM SUMMATION FOR DIFFERENT ENVIRONMENTS IS TO BE PRINTED.
= 1, YES. (COMPLETE SHEET II-7)
= 2, NO.

IW5 = FLAG WHICH SPECIFIES WHETHER THE INPUT DATA WILL BE PRINTED.
= 1, YES.
= 2, NO.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET II-2
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______

69 70	71 73	77 80
15	000	16PA
R R	CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
M5					
Segm 1	3398		2		
2	3399		7		
3	3400				
4	3401				
5	3402				
6	3403				
7	3404				
8	3405				
9	3406				
10	3407				
11	3408				
12	3409				
13	3410				
14	3411				
15	3412				
16	3413				
17	3414				
18	3415				
19	3416				
20	3417		7		

QUAN	LOC	±	VALUE	±	E
M5					
Segm 21	3418		7		
22	3419				
23	3420				
24	3421				
25	3422				
26	3423				
27	3424				
28	3425				
29	3426				
30	3427				
31	3428				
32	3429		7		
33	3430		1		
34	3431		2		
35	3432				
36	3433				
37	3434				
38	3435				
39	3436				
40	3437				

M5 = FLAG WHICH SELECTS THE SPECTRUM INCREMENTAL LOAD (Δy) INPUT FORMAT. A VALUE (1 TO 7) MUST BE ENTERED FOR EVERY SEGMENT BEING ANALYZED, EXCEPT FOR THOSE SEGMENTS WITH M3 = 10, 11 OR 12.

= 1 TO 6; Δy TABLES 1 TO 6.

= 7; Δy CALCULATED BY EQUATION $\Delta y = \Delta y_1 + \Delta y_{11} (j - 1.0)$, $j = 1, 2, 3 \ldots N$.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET II-3
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date__________

69 70	71 73	77 80
1 5	0 0 0	1 6 P A
R R	CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
ISTRES					
Segm 1	6		0		
2	7				
3	8				
4	9				
5	10				
6	11				
7	12				
8	13				
9	14				
10	15				
11	16				
12	17				
13	18				
14	19				
15	20				
16	21				
17	22				
18	23				
19	24				
20	25		0		

QUAN	LOC	±	VALUE	±	E
ISTRES					
Segm 21	26		0		
22	27				
23	28				
24	29				
25	30				
26	31				
27	32				
28	33				
29	34				
30	35				
31	36				
32	37				
33	38				
34	39		0		
35	40				
36	41				
37	42				
38	43				
39	44				
40	45				

ISTRES = FLAG WHICH SPECIFIES WHETHER STRESS TABLES WILL BE USED OR NOT.

= 0, NO.

= 1 TO 14, YES; A VALUE, 1 TO 14, CORRESPONDS TO THE STRESS TABLE NUMBER TO BE USED. (STRESS TABLES MAY BE USED ONLY WHEN M3 = 1 TO 9, OR 13 TO 15.)

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET II-4
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______

69 70 71 73	77 80
1 5 0 0 0	1 6 P A
R R CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
N1FLAG					
Segm 1	3438		2		
2	3439		2		
3	3440		2		
4	3441		1		
5	3442		2		
6	3443				
7	3444				
8	3445				
9	3446				
10	3447				
11	3448		2		
12	3449		1		
13	3450		2		
14	3451		2		
15	3452		1		
16	3453		2		
17	3454				
18	3455				
19	3456				
20	3457		2		
Segm 21	3458		2		
22	3459		2		
23	3460		1		
24	3461		2		
25	3462		2		
26	3463		1		
27	3464		2		
28	3465		2		
29	3466		1		
30	3467		2		
31	3468		2		
32	3469		1		
33	3470		3		
34	3471		2		
35	3472				
36	3473				
37	3474				
38	3475				
39	3476				
40	3477				

N1FLAG = FLAG WHICH SPECIFIES THE LOAD CYCLE FORMAT.

= 1, F $[(Y_c + \Delta y/2) \pm \Delta y/2] + P$

= 2, F $[Y_c \pm \Delta y] + P$

= 3, F $[(Y_c - \Delta y/2) \pm \Delta y/2] + P$

A VALUE MUST BE INPUT FOR EVERY SEGMENT BEING UTILIZED, EXCEPT FOR SEGMENTS WITH M3 = 10, 11 OR 12.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET II-5
STANDARD DATA INPUT 1

Page____of________
Prepared by________
Date________________

69 70	71 73	77 80
1 5	0 0 0	1 6 P A
R R	CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E	QUAN	LOC	±	VALUE	±	E
IA						IA					
Segm 1	1633					Segm 21	1653				
2	1634					22	1654				
3	1635					23	1655				
4	1636					24	1656				
5	1637					25	1657				
6	1638					26	1658				
7	1639					27	1659				
8	1640					28	1660				
9	1641					29	1661				
10	1642					30	1662				
11	1643					31	1663				
12	1644					32	1664				
13	1645					33	1665				
14	1646					34	1666				
15	1647					35	1667				
16	1648					36	1668				
17	1649					37	1669				
18	1650					38	1670				
19	1651					39	1671				
20	1652					40	1672				

IA = FLAG WHICH SPECIFIES THE S-N DATA TO BE USED.
A VALUE CORRESPONDING TO THE S-N DATA TABLE NUMBER AND DATA FORMAT AS DESCRIBED BELOW MUST BE ENTERED FOR EVERY SEGMENT BEING ANALYZED.

IA	X-ARGUMENT	Y-ARGUMENT
1 TO 6	S_{MAX}/S_{ULT}	$(S_{MIN}/S_{MAX}) = R$
7 TO 12	S_{MAX}/S_{ULT}	S_{MEAN}/S_{ULT}
13 TO 18	S_{ALT}/S_{ULT}	S_{MEAN}/S_{ULT}
19 TO 24	S_{MAX}/S_{ULT}	S_{MIN}/S_{ULT}

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-1.1
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date____________

69 70 (R R)	71 73 (CASE)	77 80 (PROG)
15	000	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
AM					
Segm 1	1753	1			01
2	1754				
3	1755				
4	1756				
5	1757				
6	1758				
7	1759				
8	1760				
9	1761				
10	1762				
11	1763				
12	1764				
13	1765				
14	1766				
15	1767				
16	1768				
17	1769				
18	1770				
19	1771				
20	1772	1			01

QUAN	LOC	±	VALUE	±	E
AM					
Segm 21	1773	1			01
22	1774				
23	1775				
24	1776				
25	1777				
26	1778				
27	1779				
28	1780				
29	1781				
30	1782				
31	1783				
32	1784				
33	1785				
34	1786	1			01
35	1787				
36	1788				
37	1789				
38	1790				
39	1791				
40	1792				

AM = CONSTANT LOAD, Y_c, IN THE CYCLIC LOAD EQUATION. (MAY BE MIN., MEAN, OR MAX. LOAD DEPENDING ON THE LOAD CYCLE FORMAT.)
A VALUE MUST BE ENTERED FOR EVERY SEGMENT BEING ANALYZED, EXCEPT WHEN M3 = 10 TO 12.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-3.1
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date__________

69 70 (RR)	71 73 (CASE)	77 80 (PROG)
15	000	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
Δy_1					
Segm 1	256	1		1	
2	257	1	1	1	00
3	258	1		1	
4	259	1		1	
5	260	1		1	
6	261	1		1	
7	262	1		1	
8	263	1		1	
9	264	1		1	
10	265	1		1	
11	266	1		1	
12	267	1		1	
13	268	1		1	
14	269	1		1	
15	270	1		1	
16	271	1		1	
17	272	1		1	
18	273	1		1	
19	274	1		1	
20	275	1	1	1	00

QUAN	LOC	±	VALUE	±	E
Δy_1					
Segm 21	276	1	1	1	00
22	277	1		1	
23	278	1		1	
24	279	1		1	
25	280	1		1	
26	281	1		1	
27	282	1		1	
28	283	1		1	
29	284	1		1	
30	285	1		1	
31	286	1		1	
32	287	1	1	1	00
33	288	1		1	
34	289	1		1	
35	290	1		1	
36	291	1		1	
37	292	1		1	
38	293	1		1	
39	294	1		1	
40	295	1		1	

Δy_1 = FIRST TERM IN THE INCREMENTAL LOAD EQUATION $\Delta y = \Delta y_1 + \Delta y_{11} (j - 1.0)$
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M5 = 7.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-3.2
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______

69	70 71	73 77	80
1 5	0 0 0	1 6 P A	
R R	CASE	PROG	

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
Δy_{11}					
Segm 1	296				
2	297	1			10 0
3	298				
4	299				
5	300				
6	301				
7	302				
8	303				
9	304				
10	305				
11	306				
12	307				
13	308				
14	309				
15	310				
16	311				
17	312				
18	313				
19	314				
20	315	1			10 0
Segm 21	316	1			10 0
22	317				
23	318				
24	319				
25	320				
26	321				
27	322				
28	323				
29	324				
30	325				
31	326				
32	327	1			10 0
33	328				
34	329				
35	330				
36	331				
37	332				
38	333				
39	334				
40	335				

Δy_{11} = SECOND TERM IN THE INCREMENTAL LOAD EQUATION $\Delta y = \Delta y_1 + \Delta y_{11}$ (j-1.0).
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M5 = 7.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-4.1
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______________

69 70	71 73	77 80
1 5	0 0 0	1 6 P A
R R	CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E	QUAN	LOC	± VALUE	± E
N_{O_1}				N_{O_1}			
Segm 1	336			Segm 21	356	1806	02
2	337	115	02	22	357	883	04
3	338	9000	04	23	358	35	06
4	339	260000	06	24	359	1806	02
5	340	1806	02	25	360	883	04
6	341			26	361	35	06
7	342			27	362	1806	02
8	343			28	363	883	04
9	344			29	364	35	06
10	345	1806	02	30	365	115	02
11	346	9000	04	31	366	883	04
12	347	260000	06	32	367	35	06
13	348	1806	02	33	368		
14	349	452	03	34	369		
15	350	9	05	35	370		
16	351	1806	02	36	371		
17	352			37	372		
18	353			38	373		
19	354			39	374		
20	355	1806	02	40	375		

N_{O_1} = Σn AT Δy = 0 FOR THE FIRST TERM OF THE GENERAL OR GUST EQUATION SPECTRUM Σn INPUT.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 7 TO 9, OR 13 TO 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-4.2
STANDARD DATA INPUT 1

69 70 R	71 73 CASE	77 80 PROG
15	000	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E	QUAN	LOC	± VALUE	± E
N_{O_2}				N_{O_2}			
Segm 1	376			Segm 21	396	11898	102
2	377	11725	102	22	397	122	103
3	378	11	104	23	398	14	103
4	379	154	102	24	399	11898	102
5	380	11898	102	25	400	122	103
6	381			26	401	14	103
7	382			27	402	11898	102
8	383			28	403	122	103
9	384			29	404	14	103
10	385	11898	102	30	405	11725	102
11	386	11	104	31	406	122	103
12	387	154	102	32	407	14	103
13	388	11898	102	33	408		
14	389	195	102	34	409		
15	390	175	102	35	410		
16	391	11898	102	36	411		
17	392			37	412		
18	393			38	413		
19	394			39	414		
20	395	11898	102	40	415		

N_{O_2} = Σn AT Δy FOR THE SECOND TERM OF THE GENERAL OR GUST EQUATION SPECTRUM Σn INPUT. ENTER N_{O_2} = 0 IF SECOND TERM IS NOT USED.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 7 TO 9, OR 13 TO 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-4.3
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______

69-70 RR	71-73 CASE	77-80 PROG
15	000	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
N_{O_3}					
Segm 1	416				
2	417				
3	418		10		0
4	419				
5	420				
6	421				
7	422				
8	423				
9	424				
10	425				
11	426		10		0
12	427				
13	428				
14	429		10		0
15	430				
16	431				
17	432				
18	433				
19	434				
20	435				

QUAN	LOC	±	VALUE	±	E
N_{O_3}					
Segm 21	436				
22	437		10		0
23	438				
24	439				
25	440		10		0
26	441				
27	442				
28	443		10		0
29	444				
30	445				
31	446		10		0
32	447				
33	448				
34	449				
35	450				
36	451				
37	452				
38	453				
39	454				
40	455				

N_{O_3} = Σn AT Δy = 0 FOR THE THIRD TERM OF GENERAL EQUATION SPECTRUM Σn INPUT.
ENTER N_{O_3} = 0 IF THIRD TERM IS NOT USED.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 7.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-5.1
STANDARD DATA INPUT 1

Page____ of______
Prepared by______
Date________

R R (69–70)	CASE (71–73)	PROG (77–80)
15	000	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
$\sigma_{\Delta y_1}$					
Segm 1	456				
2	457				
3	458		12		00
4	459				
5	460				
6	461				
7	462				
8	463				
9	464				
10	465				
11	466		12		00
12	467				
13	468				
14	469		1452		00
15	470				
16	471				
17	472				
18	473				
19	474				
20	475				

QUAN	LOC	±	VALUE	±	E
$\sigma_{\Delta y_1}$					
Segm 21	476				
22	477		1336		00
23	478				
24	479				
25	480		1336		00
26	481				
27	482				
28	483		1336		00
29	484				
30	485				
31	486		1336		06
32	487				
33	488				
34	489				
35	490				
36	491				
37	492				
38	493				
39	494				
40	495				

$\sigma_{\Delta y_1}$ = RMS OF Δy IN THE FIRST TERM OF THE SPECTRUM GENERAL EQUATION.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 7.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-5.2
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date__________

69 70 RR	71 73 CASE	77 80 PROG
15	000	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E	QUAN	LOC	± VALUE	± E
$\sigma_{\Delta y_2}$				$\sigma_{\Delta y_2}$			
Segm 1	496			Segm 21	516		
2	497			22	517	1827	00
3	498	1614	00	23	518		
4	499			24	519		
5	500			25	520	1827	00
6	501			26	521		
7	502			27	522		
8	503			28	523	1827	00
9	504			29	524		
10	505			30	525		
11	506	1614	00	31	526	1827	00
12	507			32	527		
13	508			33	528		
14	509	1885	00	34	529		
15	510			35	530		
16	511			36	531		
17	512			37	532		
18	513			38	533		
19	514			39	534		
20	515			40	535		

$\sigma_{\Delta y_2}$ = RMS OF Δy IN THE SECOND TERM OF THE SPECTRUM GENERAL EQUATION. ENTER $\sigma_{\Delta y_2} = \sigma_{\Delta y_1}$ IF SECOND TERM IS NOT USED.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 7.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-5.3
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date__________

RR (69–70)	CASE (71–73)	PROG (77–80)
15	000	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
$\sigma_{\Delta y_3}$					
Segm 1	536				
2	537				
3	538		1		00
4	539				
5	540				
6	541				
7	542				
8	543				
9	544				
10	545				
11	546		1		00
12	547				
13	548				
14	549		1		00
15	550				
16	551				
17	552				
18	553				
19	554				
20	555				

QUAN	LOC	±	VALUE	±	E
$\sigma_{\Delta y_3}$					
Segm 21	556				
22	557		1		00
23	558				
24	559				
25	560		1		00
26	561				
27	562				
28	563		1		00
29	564				
30	565				
31	566		1		00
32	567				
33	568				
34	569				
35	570				
36	571				
37	572				
38	573				
39	574				
40	575				

$\sigma_{\Delta y_3}$ = RMS OF Δy IN THE THIRD TERM OF THE SPECTRUM GENERAL EQUATION. ENTER $\sigma_{\Delta y_3} = \sigma_{\Delta y_1}$ IF THIRD TERM IS NOT USED.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 7.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-9
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date__________

69 70 (R R)	71 73 (CASE)	77 80 (PROG)
15	000	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
$\bar{A}$					
Segm 1	1593				
2	1594				
3	1595				01
4	1596				01
5	1597				
6	1598				
7	1599				
8	1600				
9	1601				
10	1602				
11	1603				01
12	1604				01
13	1605				
14	1606				01
15	1607				01
16	1608				
17	1609				
18	1610				
19	1611				
20	1612				

QUAN	LOC	±	VALUE	±	E
$\bar{A}$					
Segm 21	1613				
22	1614				01
23	1615				01
24	1616				
25	1617				01
26	1618				01
27	1619				
28	1620				01
29	1621				01
30	1622				
31	1623				01
32	1624				01
33	1625				
34	1626				
35	1627				
36	1628				
37	1629				
38	1630				
39	1631				
40	1632				

$\bar{A}$ = GUST RESPONSE FACTOR $= \sigma_{\Delta y}/\sigma_u$. A CONSTANT IN THE GUST EQUATION SPECTRUM INPUT.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 9.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET V-1.1
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date____________

69 70	71 73	77 80
1 5	0 0 0	1 6 P A
R R	CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
S-N DATA 1					
N_{MAX}	1855		11		08
n_Y	1856		13		01
Y Y_1	1857	-	1100		03
2	1858		10		00
3	1859		11		01
4	1860				
5	1861				
6	1862				
7	1863				
8	1864				
9	1865				
10	1866				
11	1867				
12	1868				
13	1869				
14	1870				
Y_{15}	1871				

QUAN	LOC	±	VALUE	±	E
n_X	1872		14		01
X X_1	1873		11		07
2	1874		111		07
3	1875		112		07
4	1876		113		07
5	1877				
6	1878				
7	1879				
8	1880				
9	1881				
10	1882				
11	1883				
12	1884				
13	1885				
14	1886				
X_{15}	1887				

S-N DATA TABLE 1; N = f(Y,X) = CYCLES TO FAILURE
(Y,X) = LOADS, SEE DATA SHEET II-5 FOR LOADS FORMATS.

N_{MAX} = MAXIMUM N CONSIDERED FOR DAMAGE CALCULATION, CYCLES

n_Y = NUMBER OF Y ENTRIES, $2 \leq n_Y \leq 15$.

n_X = NUMBER OF X ENTRIES, $4 \leq n_X \leq 15$.

Y = S-N CURVE CONSTANT LOAD, ENTER Y VALUES IN ASCENDING ORDER.

X = S-N CURVE VARIABLE LOAD, ENTER X VALUES IN ASCENDING ORDER.

THE ABOVE VALUES AND THE CORRESPONDING N VALUES, TO BE ENTERED ON SUBSEQUENT LOAD SHEETS, MUST BE INPUT ONLY WHEN IA AND/OR I4 = 1, 7, 13 OR 19 IS USED.

Page____of____
Prepared by____
Date ____

RR (69–70)	CASE (71–73)	PROG (77–80)
15	001	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
IRR	201		15		X
ICASE	202		1		X
IEND	1		34		X
I4	3		0		X
KEND	2				X
NEND	1833				
S_{ULT}	4		1		01
S	5		21072		04
$\bar{c}$	196		1787		02

IRR = REFERENCE RUN NUMBER.
ICASE = CASE NUMBER.
IRR AND ICASE NEED TO BE ENTERED ONLY IF ANY OF THE OPTIONAL PRINTOUT FLAGS IW1, IW2, IW4, IW5 = 1.
IEND = NUMBER OF SEGMENTS IN THE CASE ($1 \leq$ IEND ≤ 40).
I4 = FLAG WHICH SPECIFIES WHETHER GAG DAMAGE IS TO BE CALCULATED.
= 0, NO.
= 1 TO 24, YES. THE NUMBER CORRESPONDS TO THE S-N DATA TABLE NUMBER TO BE USED IN GAG DAMAGE CALCULATION.
KEND = NUMBER OF THE LAST SEGMENT TO BE USED IN THE GAG CYCLE SPECTRUM DEFINITION. ENTER ONLY IF I4 $\neq$ 0.
NEND = NUMBER OF GAG CYCLES TO BE DEFINED FROM THE INPUT SPECTRUM. ENTER ONLY IF I4 $\neq$ 0.
S_{ULT} = STRUCTURAL ELEMENT ULTIMATE STATIC STRENGTH.
S = WING AREA, FT^2. ENTER ONLY IF M3 = 8 OR 13 IN ANY SEGMENT.
$\bar{c}$ = WING MEAN AERODYNAMIC CHORD, FT. ENTER ONLY IF M3 = 13 IN ANY SEGMENT.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET II-1
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date____________

69 70 R R	71 73 CASE	77 80 PROG
15	001	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E	QUAN	LOC	±	VALUE	±	E
M3						M3					
Segm 1	1673		2			Segm 21	1693		13		
2	1674		13			22	1694		9		
3	1675		9			23	1695		9		
4	1676		9			24	1696		13		
5	1677		13			25	1697		9		
6	1678					26	1698		9		
7	1679					27	1699		13		
8	1680					28	1700		9		
9	1681					29	1701		9		
10	1682		13			30	1702		13		
11	1683		9			31	1703		9		
12	1684		9			32	1704		9		
13	1685		13			33	1705		1		
14	1686		9			34	1706		2		
15	1687		9			35	1707				
16	1688		13			36	1708				
17	1689					37	1709				
18	1690					38	1710				
19	1691					39	1711				
20	1692		13			40	1712				

M3 = FLAG WHICH SELECTS THE SPECTRUM CYCLE INPUT FORMAT; A VALUE (1 TO 15) MUST BE ENTERED FOR EVERY SEGMENT BEING ANALYZED.

= 1 TO 6; Σn TABLES 1 TO 6.

= 7; GENERAL EQUATION, $\Sigma n = (\Sigma N_{0i} e^{-(\Delta y)^2/2(\sigma_{\Delta y})_i^2})T$, $i = 1, 2, 3$

= 8; GUST EQUATION, $\Sigma n = (\Sigma N_{0i} P_i e^{-\Delta y/b_i \bar{A}})T$, $i = 1, 2$; $\bar{A}$ IS CALCULATED BY THE PROGRAM FOR AIRPLANE C. G. VERTICAL GUST LOAD FACTOR.

= 9; GUST EQUATION AS ABOVE; $\bar{A}$ IS DIRECTLY INPUT.

= 10 TO 12; (S_{MAX}, S_{MIN}, n) TABLES 1 TO 3.

= 13; GUST EQUATION AS ABOVE; K_{σ_u} AND $\bar{A}$ ARE CALCULATED BY THE PROGRAM FOR AIRPLANE C. G. VERTICAL GUST LOAD FACTOR.

= 14; GUST EQUATION AS ABOVE; $\bar{A}$ IS CALCULATED BY THE PROGRAM FOR AIRPLANE C. G. SIDE GUST LOAD FACTOR.

= 15; GUST EQUATION AS ABOVE; $\bar{A}$ IS CALCULATED BY THE PROGRAM FOR VERTICAL TAIL SIDE LOAD.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-1.2
STANDARD DATA INPUT 1

Page____ of______
Prepared by______
Date________

RR (69-70)	CASE (71-73)	PROG (77-80)
15	001	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
F					
Segm 1	3478	-	8736		04
2	3479		8043		
3	3480		8043		
4	3481		8043		
5	3482		8415		
6	3483				
7	3484				
8	3485				
9	3486				
10	3487				
11	3488				
12	3489		8415		
13	3490		8828		
14	3491		8828		
15	3492		8828		
16	3493		8927		
17	3494				
18	3495				
19	3496				
20	3497		8927		04

QUAN	LOC	±	VALUE	±	E
F					
Segm 21	3498		8927		04
22	3499		8927		
23	3500		8927		
24	3501		8877		
25	3502				
26	3503				
27	3504				
28	3505				
29	3506				
30	3507				
31	3508				
32	3509		8877		
33	3510		8910		
34	3511	-	6765		04
35	3512				
36	3513				
37	3514				
38	3515				
39	3516				
40	3517				

F = A MULTIPLICATION FACTOR IN THE CYCLIC LOAD EQUATIONS.
A VALUE MUST BE ENTERED FOR EVERY SEGMENT BEING UTILIZED, EXCEPT FOR SEGMENTS WITH M3 = 10, 11, OR 12, ONLY WHEN L1 = 1.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-1.3
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date____________

69 70	71 73	77 80
15	001	16PA
R R	CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
P			
Segm 1	3598	10	100
2	3599	1413	103
3	3600	1413	103
4	3601	1413	103
5	3602	-11716	104
6	3603	1	1
7	3604	1	1
8	3605	1	1
9	3606	1	1
10	3607	1	1
11	3608	1	1
12	3609	-11716	1
13	3610	-12021	1
14	3611	-12021	1
15	3612	-12021	1
16	3613	-11625	1
17	3614	1	1
18	3615	1	1
19	3616	1	1
20	3617	-11625	104

QUAN	LOC	± VALUE	± E
P			
Segm 21	3618	-11625	104
22	3619	-11625	104
23	3620	-11625	104
24	3621	1239	103
25	3622	1239	1
26	3623	1239	1
27	3624	1322	1
28	3625	1322	1
29	3626	1322	1
30	3627	-1256	1
31	3628	-1256	1
32	3629	-1256	103
33	3630	-13805	104
34	3631	10	100
35	3632	1	1
36	3633	1	1
37	3634	1	1
38	3635	1	1
39	3636	1	1
40	3637	1	1

P = A CONSTANT IN THE CYCLE LOADS EQUATIONS.
A VALUE MUST BE ENTERED FOR EVERY SEGMENT BEING UTILIZED, EXCEPT FOR SEGMENTS WITH M3 = 10, 11, OR 12, ONLY WHEN IW3 = 1.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-1.4
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______

69 70	71 73	77 80
15	001	16PA
R R	CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN N	LOC	±	VALUE	±	E
Segm 1	1793		8		
2	1794		10		
3	1795		12		
4	1796		14		
5	1797		10		
6	1798				
7	1799				
8	1800				
9	1801				
10	1802		10		
11	1803		12		
12	1804		14		
13	1805		10		
14	1806		10		
15	1807		18		
16	1808		10		
17	1809				
18	1810				
19	1811				
20	1812		10		

QUAN N	LOC	±	VALUE	±	E
Segm 21	1813		10		
22	1814		8		
23	1815		24		
24	1816		10		
25	1817		8		
26	1818		24		
27	1819		10		
28	1820		8		
29	1821		24		
30	1822		10		
31	1823		8		
32	1824		24		
33	1825		22		
34	1826		8		
35	1827				
36	1828				
37	1829				
38	1830				
39	1831				
40	1832				

N = NUMBER OF LOAD LEVELS ENTERED IN THE LOAD SPECTRUM FOR ONE SEGMENT. ($2 \leq N \leq 25$) WHEN M3 = 1 TO 9 OR 13 TO 15; ($1 \leq n \leq 25$) WHEN M3 = 10, 11 OR 12.
A VALUE MUST BE ENTERED FOR EVERY SEGMENT BEING ANALYZED.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-1.5
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date__________

69 70 (R R)	71 73 (CASE)	77 80 (PROG)
1 5	0 0 1	1 6 P A

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
T			
Segm 1	576	11112	101
2	577	115447	104
3	578	101334	100
4	579	101334	100
5	580	122361	104
6	581	129235	104
7	582	167043	104
8	583	1178243	105
9	584	1272096	105
10	585	1504848	105
11	586	130489	100
12	587	130489	100
13	588	1196080	106
14	589	14982	100
15	590	14982	100
16	591	119418	105
17	592	126599	105
18	593	124197	105
19	594	111520	105
20	595	14428	104

QUAN	LOC	± VALUE	± E
T			
Segm 21	596	10	100
22	597	12547	100
23	598	12547	100
24	599	12847	104
25	600	101658	100
26	601	101658	100
27	602	116781	104
28	603	101213	100
29	604	101213	100
30	605	111686	104
31	606	101132	100
32	607	101132	100
33	608	11112	101
34	609	11112	101
35	610	1	1
36	611	1	1
37	612	1	1
38	613	1	1
39	614	1	1
40	615	1	1

T = SPECTRUM, Σn, MULTIPLICATION FACTOR.
A VALUE MUST BE ENTERED FOR EVERY SEGMENT BEING UTILIZED, EXCEPT FOR SEGMENTS WITH M3 = 10, 11, OR 12.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-2.1
STANDARD DATA INPUT 1

Page ____ of ____
Prepared by ____
Date ____

69 70 (R R)	71 73 (CASE)	77 80 (PROG)
15	001	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
Δy TABLE 1					
j = 1	46		1		00
2	47		2		
3	48		3		
4	49		4		
5	50		5		
6	51		6		
7	52		7		
8	53		8		00
9	54		9		01
10	55		1		
11	56		12		
12	57		14		
13	58		16		
14	59		18		
15	60		2		
16	61		22		
17	62		24		
18	63		26		
19	64		28		
20	65		3		
21	66		32		
22	67		34		01
23	68				
24	69				
25	70				

QUAN	LOC	±	VALUE	±	E
Σn TABLE 1					
j = 1	71		1000		04
2	72		900		03
3	73		600		
4	74		450		
5	75		320		
6	76		250		
7	77		210		
8	78		163		
9	79		133		
10	80		110		03
11	81		70		02
12	82		50		
13	83		37		
14	84		26		
15	85		20		
16	86		155		
17	87		113		02
18	88		74		01
19	89		48		
20	90		32		
21	91		22		01
22	92		0		00
23	93				
24	94				
25	95				

INPUT SPECTRUM Δy — Σn TABLE NO. 1.
A MAXIMUM OF 25 VALUES MAY BE ENTERED. AT LEAST TWO VALUES MUST BE ENTERED.
Δy VALUES MUST BE ENTERED ONLY IF M5 = 1 IS USED.
Σn VALUES MUST BE ENTERED ONLY IF M3 = 1 IS USED.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-2.2
STANDARD DATA INPUT 1

Page ____ of ____
Prepared by ____
Date ____

69 70 (R R)	71 73 (CASE)	77 80 (PROG)
15	001	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN (Δy TABLE 2)	LOC	±	VALUE	±	E
j = 1	231		3		00
2	232		4		
3	233		5		
4	234		6		
5	235		7		
6	236		8		
7	237		9		00
8	238		1		01
9	239				
10	240				
11	241				
12	242				
13	243				
14	244				
15	245				
16	246				
17	247				
18	248				
19	249				
20	250				
21	251				
22	252				
23	253				
24	254				
25	255				

QUAN (Σn TABLE 2)	LOC	±	VALUE	±	E
j = 1	96		5000		04
2	97		5000		04
3	98		750		03
4	99		100		03
5	100		1.5		02
6	101		2		01
7	102		3		00
8	103		0.5		00
9	104				
10	105				
11	106				
12	107				
13	108				
14	109				
15	110				
16	111				
17	112				
18	113				
19	114				
20	115				
21	116				
22	117				
23	118				
24	119				
25	120				

INPUT SPECTRUM $\Delta y - \Sigma n$ TABLE NO. 2.
A MAXIMUM OF 25 VALUES MAY BE ENTERED. AT LEAST TWO VALUES MUST BE ENTERED.
Δy VALUES MUST BE ENTERED ONLY IF M5 = 2 IS USED.
Σn VALUES MUST BE ENTERED ONLY IF M3 = 2 IS USED.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-4.1
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date__________

69-70 R R	71-73 CASE	77-80 PROG
15	001	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
N_{O_1}					
Segm 1	336				
2	337				
3	338		27		06
4	339		9		05
5	340				
6	341				
7	342				
8	343				
9	344				
10	345				
11	346		27		06
12	347		9		05
13	348				
14	349		4		05
15	350		21		05
16	351				
17	352				
18	353				
19	354				
20	355				

QUAN	LOC	±	VALUE	±	E
N_{O_1}					
Segm 21	356				
22	357		19		06
23	358		17		06
24	359				
25	360		19		06
26	361		17		06
27	362				
28	363		19		06
29	364		17		06
30	365				
31	366		19		06
32	367		17		06
33	368				
34	369				
35	370				
36	371				
37	372				
38	373				
39	374				
40	375				

N_{O_1} = Σn AT Δy = 0 FOR THE FIRST TERM OF THE GENERAL OR GUST EQUATION SPECTRUM Σn INPUT.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 7 TO 9, OR 13 TO 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-4.2
STANDARD DATA INPUT 1

Page____ of______
Prepared by______
Date______

69 70	71 73	77 80
15	001	16PA
R R	CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E	QUAN	LOC	± VALUE	± E
N_{O_2}				N_{O_2}			
Segm 1	376			Segm 21	396		
2	377			22	397	10	00
3	378	125	03	23	398	145	03
4	379	150	02	24	399		
5	380			25	400	10	00
6	381			26	401	145	03
7	382			27	402		
8	383			28	403	10	00
9	384			29	404	145	03
10	385			30	405		
11	386	125	03	31	406	10	00
12	387	150	02	32	407	145	03
13	388			33	408		
14	389	11	04	34	409		
15	390	162	02	35	410		
16	391			36	411		
17	392			37	412		
18	393			38	413		
19	394			39	414		
20	395			40	415		

N_{O_2} = Σn AT Δy FOR THE SECOND TERM OF THE GENERAL OR GUST EQUATION SPECTRUM Σn INPUT. ENTER N_{O_2} = 0 IF SECOND TERM IS NOT USED.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 7 TO 9, OR 13 TO 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-6.2
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______

RR (69-70)	CASE (71-73)	PROG (77-80)
15	001	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
m			
Segm 1	656		
2	657	715	01
3	658		
4	659		
5	660	642	01
6	661	643	
7	662	648	
8	663	662	
9	664	687	
10	665	732	01
11	666		
12	667		
13	668	77	01
14	669		
15	670		
16	671	77	01
17	672	707	
18	673	656	
19	674	641	
20	675	635	01

QUAN	LOC	± VALUE	± E
m			
Segm 21	676	635	01
22	677		
23	678		
24	679	635	01
25	680		
26	681		
27	682	657	01
28	683		
29	684		
30	685	637	01
31	686		
32	687		
33	688		
34	689		
35	690		
36	691		
37	692		
38	693		
39	694		
40	695		

m = WING LIFT CURVE SLOPE PER RADIAN.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8, OR 13.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-6.3
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date____________

69 70	71 73	77 80
1 5	0 0 1	1 6 P A
R R	CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
V_e			
Segm 1	696		
2	697	115	03
3	698		
4	699		
5	700	2815	03
6	701	2781	
7	702	2694	
8	703	2594	
9	704	2373	
10	705	216	03
11	706		
12	707		
13	708	205	03
14	709		
15	710		
16	711	225	03
17	712	250	
18	713		
19	714		
20	715	250	03

QUAN	LOC	± VALUE	± E
V_e			
Segm 21	716	250	03
22	717		
23	718		
24	719	165	03
25	720		
26	721		
27	722	135	03
28	723		
29	724		
30	725	105	03
31	726		
32	727		
33	728		
34	729		
35	730		
36	731		
37	732		
38	733		
39	734		
40	735		

V_e = AIRPLANE SPEED, (EQUIVALENT AIRSPEED), KNOTS.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8 OR 13.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-6.4
STANDARD DATA INPUT 1

Page____ of______
Prepared by______
Date______

69 70	71 73	77 80
15	001	16PA
R R	CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
W			
Segm 1	736		
2	737	166021	06
3	738		
4	739		
5	740	163109	06
6	741		
7	742		
8	743		
9	744		
10	745	163109	06
11	746		
12	747		
13	748	158677	06
14	749		
15	750		
16	751	156213	06
17	752		
18	753		
19	754		
20	755	156213	06

QUAN	LOC	± VALUE	± E
W			
Segm 21	756	156213	06
22	757		
23	758		
24	759	156213	06
25	760		
26	761		
27	762	156213	06
28	763		
29	764		
30	765	156213	06
31	766		
32	767		
33	768		
34	769		
35	770		
36	771		
37	772		
38	773		
39	774		
40	775		

W = AIRPLANE GROSS WEIGHT, LBS.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8 OR 13.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-6.5
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______________

69 70	71 73	77 80
1 5	0 0 1	1 6 P A
R R	CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
SIG			
Segm 1	1713		
2	1714	1985	100
3	1715		
4	1716		
5	1717	1959	100
6	1718	1908	
7	1719	1798	
8	1720	1629	
9	1721	1448	
10	1722	1326	100
11	1723		
12	1724		
13	1725	1271	100
14	1726		
15	1727		
16	1728	1322	100
17	1729	1448	
18	1730	1629	
19	1731	1798	
20	1732	1888	100

QUAN	LOC	± VALUE	± E
SIG			
Segm 21	1733	1943	100
22	1734		
23	1735		
24	1736	1935	100
25	1737		
26	1738		
27	1739	1971	100
28	1740		
29	1741		
30	1742	1993	100
31	1743		
32	1744		
33	1745		
34	1746		
35	1747		
36	1748		
37	1749		
38	1750		
39	1751		
40	1752		

SIG = AIR DENSITY RATIO = ρ/ρ_0.
A VALUE MUST BE ENTERED ONLY FOR THOSE SEGMENTS WITH M3 = 13.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-6.6
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date__________

69 70 (R R)	71–73 (CASE)	77–80 (PROG)
15	001	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
L or l_t					
Segm 1	3638				
2	3639		500		03
3	3640				
4	3641				
5	3642		1750		04
6	3643		2500		
7	3644				
8	3645				
9	3646				
10	3647		2500		04
11	3648				
12	3649				
13	3650		2500		04
14	3651				
15	3652				
16	3653		2500		04
17	3654				
18	3655				
19	3656				
20	3657		2500		04

QUAN	LOC	±	VALUE	±	E
L or l_t					
Segm 21	3658		1920		04
22	3659				
23	3660				
24	3661		2050		04
25	3662				
26	3663				
27	3664		1150		04
28	3665				
29	3666				
30	3667		500		03
31	3668				
32	3669				
33	3670				
34	3671				
35	3672				
36	3673				
37	3674				
38	3675				
39	3676				
40	3677				

L = SCALE OF TURBULENCE, FT.
A VALUE MUST BE ENTERED ONLY FOR THOSE SEGMENTS WITH M3 = 13.

l_t = DISTANCE FROM AIRPLANE C.G. TO LIFT CENTER OF VERTICAL SURFACE, INCHES.
A VALUE MUST BE ENTERED ONLY FOR THOSE SEGMENTS WITH M3 = 14 or 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-7.1
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date __________

69 RR	70 71 CASE 73	77 PROG 80
15	001	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
P_1			
Segm 1	776	1	1
2	777	11	10,1
3	778	11	10,1
4	779	11	10,1
5	780	14,2	10,0
6	781	13	1
7	782	11,5	1
8	783	1062	1
9	784	10,2,5	1
10	785	1012	10,0
11	786	11	10,1
12	787	11	10,1
13	788	10085	10,0
14	789	11	10,1
15	790	11	10,1
16	791	10,12	10,0
17	792	10,2,5	1
18	793	10,6,2	1
19	794	11,5	1
20	795	12,9	10,0

QUAN	LOC	± VALUE	± E
P_1			
Segm 21	796	14	10,0
22	797	11	10,1
23	798	11	10,1
24	799	13,9	10,0
25	800	11	10,1
26	801	11	10,1
27	802	16,2	10,0
28	803	11	10,1
29	804	1	1
30	805	1	1
31	806	1	1
32	807	11	10,1
33	808	1	1
34	809	1	1
35	810	1	1
36	811	1	1
37	812	1	1
38	813	1	1
39	814	1	1
40	815	1	1

P_1 = PORTION OF FLIGHT TIME (OR DISTANCE) SPENT IN NON-STORM TURBULENCE.
THE VALUE OF P IN THE FIRST TERM OF GUST EQUATION SPECTRUM INPUT.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8, 9 OR 13 TO 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-7.2
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date____________

69 70 R R	71 73 CASE	77 80 PROG
15	001	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
P_2			
Segm 1	1473		
2	1474	005	00
3	1475	1	01
4	1476	1	01
5	1477	0033	00
6	1478	002	
7	1479	00095	
8	1480	00028	
9	1481	0011	
10	1482	000095	00
11	1483	1	01
12	1484	1	01
13	1485	000098	00
14	1486	1	01
15	1487	1	01
16	1488	000095	00
17	1489	0011	
18	1490	00028	
19	1491	00095	
20	1492	0019	00

QUAN	LOC	± VALUE	± E
P_2			
Segm 21	1493	0031	00
22	1494	1	01
23	1495	1	01
24	1496	0029	00
25	1497	1	01
26	1498	1	01
27	1499	0042	00
28	1500	1	01
29	1501	1	01
30	1502	005	00
31	1503	1	01
32	1504	1	01
33	1505		
34	1506		
35	1507		
36	1508		
37	1509		
38	1510		
39	1511		
40	1512		

P_2 = PORTION OF FLIGHT TIME (OR DISTANCE) SPENT IN STORM TURBULENCE. THE VALUE OF P IN THE SECOND TERM OF GUST EQUATION SPECTRUM INPUT. ENTER P_2 = 0 IF SECOND TERM IS NOT USED.

A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8, 9 OR 13 TO 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-8.1
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date____________

69–70 (R R)	71–73 (CASE)	77–80 (PROG)
15	001	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
b_1					
Segm 1	1513				
2	1514		25		01
3	1515		0492		00
4	1516		0869		00
5	1517		293		01
6	1518		323		
7	1519		32		
8	1520		259		
9	1521		211		
10	1522		17		01
11	1523		0492		00
12	1524		0869		00
13	1525		155		01
14	1526		0523		00
15	1527		0921		00
16	1528		168		01
17	1529		211		
18	1530		259		
19	1531		32		
20	1532		325		01
Segm 21	1533		295		01
22	1534		0543		00
23	1535		0942		00
24	1536		305		01
25	1537		0543		00
26	1538		0942		00
27	1539		275		01
28	1540		0543		00
29	1541		0942		00
30	1542		25		01
31	1543		0543		00
32	1544		0942		00
33	1545				
34	1546				
35	1547				
36	1548				
37	1549				
38	1550				
39	1551				
40	1552				

b_1 = SCALE PARAMETER IN PROBABILITY DISTRIBUTION OF RMS GUST VELOCITY FOR NON-STORM TURBULENCE. THE VALUE OF b IN THE FIRST TERM OF GUST EQUATION SPECTRUM INPUT.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8, 9 OR 13 TO 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-8.2
STANDARD DATA INPUT 1

Page____ of______
Prepared by______
Date______

RR (69-70)	CASE (71-73)	PROG (77-80)
15	001	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
b_2					
Segm 1	1553				
2	1554		5		01
3	1555		1646		00
4	1556		2877		00
5	1557		575		01
6	1558		77		01
7	1559		825		01
8	1560		833		01
9	1561		794		01
10	1562		57		01
11	1563		1646		00
12	1564		2877		00
13	1565		49		01
14	1566		0986		00
15	1567		2621		00
16	1568		565		01
17	1569		794		01
18	1570		833		01
19	1571		825		01
20	1572		775		01
Segm 21	1573		6		01
22	1574		0543		00
23	1575		2311		00
24	1576		615		01
25	1577		0543		00
26	1578		2311		00
27	1579		52		01
28	1580		0543		00
29	1581		2311		00
30	1582		5		01
31	1583		0543		00
32	1584		2311		00
33	1585				
34	1586				
35	1587				
36	1588				
37	1589				
38	1590				
39	1591				
40	1592				

b_2 = SCALE PARAMETER IN PROBABILITY DISTRIBUTION OF RMS GUST VELOCITY FOR STORM TURBULENCE. THE VALUE OF b IN THE SECOND TERM OF GUST EQUATION SPECTRUM INPUT. ENTER b_2 = b_1 IF SECOND TERM IS NOT USED.

A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8, 9 OR 13 TO 15.

69 70 R R	71 73 CASE	77 80 PROG
15	002	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
IRR	201		15		
ICASE	202		2		
IEND	1		34		
I4	3		0		
KEND	2				
NEND	1833				
S_{ULT}	4		1		01
S	5		21072		04
$\bar{c}$	196		1787		02

IRR = REFERENCE RUN NUMBER.
ICASE = CASE NUMBER.
IRR AND ICASE NEED TO BE ENTERED ONLY IF ANY OF THE OPTIONAL PRINTOUT FLAGS IW1, IW2, IW4, IW5 = 1.
IEND = NUMBER OF SEGMENTS IN THE CASE ($1 \leq$ IEND ≤ 40).
I4 = FLAG WHICH SPECIFIES WHETHER GAG DAMAGE IS TO BE CALCULATED.
= 0, NO.
= 1 TO 24, YES. THE NUMBER CORRESPONDS TO THE S-N DATA TABLE NUMBER TO BE USED IN GAG DAMAGE CALCULATION.
KEND = NUMBER OF THE LAST SEGMENT TO BE USED IN THE GAG CYCLE SPECTRUM DEFINITION. ENTER ONLY IF I4 $\neq$ 0.
NEND = NUMBER OF GAG CYCLES TO BE DEFINED FROM THE INPUT SPECTRUM. ENTER ONLY IF I4 $\neq$ 0.
S_{ULT} = STRUCTURAL ELEMENT ULTIMATE STATIC STRENGTH.
S = WING AREA, FT^2. ENTER ONLY IF M3 = 8 OR 13 IN ANY SEGMENT.
$\bar{c}$ = WING MEAN AERODYNAMIC CHORD, FT. ENTER ONLY IF M3 = 13 IN ANY SEGMENT.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET II-1
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______

69	70 71	73	77	80
1	5	002	16PA	
R R		CASE	PROG	

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
M3					
Segm 1	1673		2		
2	1674		13		
3	1675		7		
4	1676		9		
5	1677		13		
6	1678				
7	1679				
8	1680				
9	1681				
10	1682		13		
11	1683		7		
12	1684		9		
13	1685		13		
14	1686		7		
15	1687		9		
16	1688		13		
17	1689				
18	1690				
19	1691				
20	1692		13		

QUAN	LOC	±	VALUE	±	E
M3					
Segm 21	1693		13		
22	1694		7		
23	1695		9		
24	1696		13		
25	1697		7		
26	1698		9		
27	1699		13		
28	1700		7		
29	1701		9		
30	1702		13		
31	1703		7		
32	1704		9		
33	1705		1		
34	1706		2		
35	1707				
36	1708				
37	1709				
38	1710				
39	1711				
40	1712				

M3 = FLAG WHICH SELECTS THE SPECTRUM CYCLE INPUT FORMAT; A VALUE (1 TO 15) MUST BE ENTERED FOR EVERY SEGMENT BEING ANALYZED.

- = 1 TO 6; Σn TABLES 1 TO 6.
- = 7; GENERAL EQUATION, $\Sigma n = (\Sigma N_{0i} e^{-(\Delta y)^2/2(\sigma_{\Delta y})_i^2})T$, $i = 1, 2, 3$
- = 8; GUST EQUATION, $\Sigma n = (\Sigma N_{0i} P_i e^{-\Delta y/b_i \bar{A}})T$, $i = 1, 2$; $\bar{A}$ IS CALCULATED BY THE PROGRAM FOR AIRPLANE C. G. VERTICAL GUST LOAD FACTOR.
- = 9; GUST EQUATION AS ABOVE; $\bar{A}$ IS DIRECTLY INPUT.
- = 10 TO 12; (S_{MAX}, S_{MIN}, n) TABLES 1 TO 3.
- = 13; GUST EQUATION AS ABOVE; K_{σ_u} AND $\bar{A}$ ARE CALCULATED BY THE PROGRAM FOR AIRPLANE C. G. VERTICAL GUST LOAD FACTOR.
- = 14; GUST EQUATION AS ABOVE; $\bar{A}$ IS CALCULATED BY THE PROGRAM FOR AIRPLANE C. G. SIDE GUST LOAD FACTOR.
- = 15; GUST EQUATION AS ABOVE; $\bar{A}$ IS CALCULATED BY THE PROGRAM FOR VERTICAL TAIL SIDE LOAD.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-1.2
STANDARD DATA INPUT 1

Page____ of______
Prepared by______
Date____________

69 70 (R R)	71 73 (CASE)	77 80 (PROG)
15	002	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
F			
Segm 1	3478	-17070	104
2	3479	112086	105
3	3480	112086	1
4	3481	112086	1
5	3482	112227	1
6	3483		
7	3484		
8	3485		
9	3486		
10	3487		
11	3488		
12	3489	112227	1
13	3490	112441	1
14	3491	112441	1
15	3492	112441	1
16	3493	112284	1
17	3494		
18	3495		
19	3496		
20	3497	112284	105

QUAN	LOC	± VALUE	± E
F			
Segm 21	3498	112284	105
22	3499	112284	1
23	3500	112284	1
24	3501	112243	1
25	3502		
26	3503		
27	3504		
28	3505		
29	3506		
30	3507		
31	3508		
32	3509	112243	105
33	3510	16765	104
34	3511	-15709	104
35	3512		
36	3513		
37	3514		
38	3515		
39	3516		
40	3517		

F = A MULTIPLICATION FACTOR IN THE CYCLIC LOAD EQUATIONS.
A VALUE MUST BE ENTERED FOR EVERY SEGMENT BEING UTILIZED, EXCEPT FOR SEGMENTS WITH M3 = 10, 11, OR 12, ONLY WHEN L1 = 1.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-1.3
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date____________

69 70	71 73	77 80
1 5	0 0 2	1 6 P A
R R	CASE	PROG

<u>KEYPUNCH</u>: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
P			
Segm 1	3598	10	100
2	3599	-11089	104
3	3600	-11089	1
4	3601	-11089	1
5	3602	-11873	1
6	3603	1	1
7	3604	1	1
8	3605	1	1
9	3606	1	1
10	3607	1	1
11	3608	1	1
12	3609	-11873	1
13	3610	-12228	1
14	3611	-12228	1
15	3612	-12228	1
16	3613	-11840	1
17	3614	-11840	1
18	3615	-11840	1
19	3616	-11840	1
20	3617	-11840	104
Segm 21	3618	-11840	104
22	3619	-11840	104
23	3620	-11840	104
24	3621	-1149	103
25	3622	-1149	1
26	3623	-1149	1
27	3624	1182	1
28	3625	1	1
29	3626	1	1
30	3627	1	1
31	3628	1	1
32	3629	1182	103
33	3630	11393	104
34	3631	10	100
35	3632	1	1
36	3633	1	1
37	3634	1	1
38	3635	1	1
39	3636	1	1
40	3637	1	1

P = A CONSTANT IN THE CYCLE LOADS EQUATIONS.
A VALUE MUST BE ENTERED FOR EVERY SEGMENT BEING UTILIZED, EXCEPT FOR SEGMENTS WITH M3 = 10, 11, OR 12, ONLY WHEN IW3 = 1.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-1.4
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______

69 70 (R R)	71 73 (CASE)	77 80 (PROG)
15	002	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
N					
Segm 1	1793		5		
2	1794		10		
3	1795		8		
4	1796		12		
5	1797		10		
6	1798				
7	1799				
8	1800				
9	1801				
10	1802		10		
11	1803		8		
12	1804		12		
13	1805		15		
14	1806		8		
15	1807		10		
16	1808				
17	1809				
18	1810				
19	1811				
20	1812		10		
Segm 21	1813		10		
22	1814		8		
23	1815		10		
24	1816		15		
25	1817		8		
26	1818		10		
27	1819		10		
28	1820		8		
29	1821		10		
30	1822		10		
31	1823		8		
32	1824		10		
33	1825		9		
34	1826		5		
35	1827				
36	1828				
37	1829				
38	1830				
39	1831				
40	1832				

N = NUMBER OF LOAD LEVELS ENTERED IN THE LOAD SPECTRUM FOR ONE SEGMENT. ($2 \le N \le 25$) WHEN M3 = 1 TO 9 OR 13 TO 15; ($1 \le n \le 25$) WHEN M3 = 10, 11 OR 12.
A VALUE MUST BE ENTERED FOR EVERY SEGMENT BEING ANALYZED.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-1.5
STANDARD DATA INPUT 1

Page____ of______
Prepared by______
Date______

69–70	71–73	77–80
15	002	16PA
R R	CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
T			
Segm 1	576	11416	101
2	577	13398	101
3	578	10243	100
4	579	10243	100
5	580	129331	104
6	581	147011	104
7	582	1105795	105
8	583	1267624	105
9	584	145312	105
10	585	1551107	105
11	586	14094	100
12	587	14094	100
13	588	12155435	106
14	589	15437	100
15	590	15437	100
16	591	1187762	105
17	592	1354566	105
18	593	1325963	105
19	594	1154627	105
20	595	163882	104

QUAN	LOC	± VALUE	± E
T			
Segm 21	596	10	100
22	597	13245	100
23	598	13245	100
24	599	172499	104
25	600	10425	100
26	601	10425	100
27	602	172216	104
28	603	10566	100
29	604	10566	1
30	605	10	1
31	606	10	1
32	607	10	100
33	608	11416	101
34	609	11416	101
35	610	1	1
36	611	1	1
37	612	1	1
38	613	1	1
39	614	1	1
40	615	1	1

T = SPECTRUM, Σn, MULTIPLICATION FACTOR.
A VALUE MUST BE ENTERED FOR EVERY SEGMENT BEING UTILIZED, EXCEPT FOR SEGMENTS WITH M3 = 10, 11, OR 12.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-2.1
STANDARD DATA INPUT 1

Page ____ of ____
Prepared by ____
Date ____

69–70 RR	71–73 CASE	77–80 PROG
15	002	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
Δy TABLE 1					
j = 1	46		1		00
2	47		2		
3	48		3		
4	49		4		
5	50		5		
6	51		6		
7	52		7		
8	53		8		
9	54		9		00
10	55				
11	56				
12	57				
13	58				
14	59				
15	60				
16	61				
17	62				
18	63				
19	64				
20	65				
21	66				
22	67				
23	68				
24	69				
25	70				

QUAN	LOC	±	VALUE	±	E
Σn TABLE 1					
j = 1	71		900		03
2	72		530		03
3	73		50		02
4	74		9		01
5	75		43		01
6	76		22		01
7	77		13		01
8	78		65		00
9	79		0		00
10	80				
11	81				
12	82				
13	83				
14	84				
15	85				
16	86				
17	87				
18	88				
19	89				
20	90				
21	91				
22	92				
23	93				
24	94				
25	95				

INPUT SPECTRUM Δy — Σn TABLE NO. 1.
A MAXIMUM OF 25 VALUES MAY BE ENTERED. AT LEAST TWO VALUES MUST BE ENTERED.
Δy VALUES MUST BE ENTERED ONLY IF M5 = 1 IS USED.
Σn VALUES MUST BE ENTERED ONLY IF M3 = 1 IS USED.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-2.2
STANDARD DATA INPUT 1

Page ____ of ____
Prepared by ____
Date ____

69	70 71	73	77	80
1	5	002	16PA	
R	R	CASE	PROG	

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN Δy TABLE 2	LOC	± VALUE	± E
j = 1	231	3	00
2	232	4	
3	233	5	
4	234	6	
5	235	7	00
6	236		
7	237		
8	238		
9	239		
10	240		
11	241		
12	242		
13	243		
14	244		
15	245		
16	246		
17	247		
18	248		
19	249		
20	250		
21	251		
22	252		
23	253		
24	254		
25	255		

QUAN Σn TABLE 2	LOC	± VALUE	± E
j = 1	96	2094	04
2	97	942	02
3	98	42	01
4	99	155	00
5	100	005	00
6	101		
7	102		
8	103		
9	104		
10	105		
11	106		
12	107		
13	108		
14	109		
15	110		
16	111		
17	112		
18	113		
19	114		
20	115		
21	116		
22	117		
23	118		
24	119		
25	120		

INPUT SPECTRUM Δy — Σn TABLE NO. 2.
A MAXIMUM OF 25 VALUES MAY BE ENTERED. AT LEAST TWO VALUES MUST BE ENTERED.
Δy VALUES MUST BE ENTERED ONLY IF M5 = 2 IS USED.
Σn VALUES MUST BE ENTERED ONLY IF M3 = 2 IS USED.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-6.2
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______________

RR (69–70)	CASE (71–73)	PROG (77–80)
15	002	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
m			
Segm 1	656		
2	657	1668	101
3	658		
4	659		
5	660	1643	101
6	661	1647	
7	662	1652	
8	663	1664	
9	664	1690	
10	665	1740	101
11	666		
12	667		
13	668	1770	101
14	669		
15	670		
16	671	1770	101
17	672	1707	
18	673	1657	
19	674	1641	
20	675	1635	101

QUAN	LOC	± VALUE	± E
m			
Segm 21	676	1635	101
22	677		
23	678		
24	679	1635	101
25	680		
26	681		
27	682	1663	101
28	683		
29	684		
30	685	1663	101
31	686		
32	687		
33	688		
34	689		
35	690		
36	691		
37	692		
38	693		
39	694		
40	695		

m = WING LIFT CURVE SLOPE PER RADIAN.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8, OR 13.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-6.3
STANDARD DATA INPUT 1

Page____ of______
Prepared by______
Date________

69 70 R R	71 73 CASE	77 80 PROG
15	002	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
V_e					
Segm 1	696				
2	697		140		03
3	698				
4	699				
5	700		2905		03
6	701		2875		
7	702		278		
8	703		2629		
9	704		2405		
10	705		2246		03
11	706				
12	707				
13	708		2184		03
14	709				
15	710				
16	711		2326		03
17	712		250		
18	713		250		
19	714		250		
20	715		250		03

QUAN	LOC	±	VALUE	±	E
V_e					
Segm 21	716		250		03
22	717				
23	718				
24	719		165		03
25	720				
26	721				
27	722		126		03
28	723				
29	724				
30	725		126		03
31	726				
32	727				
33	728				
34	729				
35	730				
36	731				
37	732				
38	733				
39	734				
40	735				

V_e = AIRPLANE SPEED, (EQUIVALENT AIRSPEED), KNOTS.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8 OR 13.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-6.4
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date__________

69 70 R R	71 73 CASE	77 80 PROG
15	002	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
W					
Segm 1	736				
2	737		190762		06
3	738				
4	739				
5	740		187441		06
6	741				
7	742				
8	743				
9	744				
10	745		187441		06
11	746				
12	747				
13	748		182588		06
14	749				
15	750				
16	751		179764		06
17	752				
18	753				
19	754				
20	755		179764		06

QUAN	LOC	±	VALUE	±	E
W					
Segm 21	756		179764		06
22	757				
23	758				
24	759		179764		06
25	760				
26	761				
27	762		179764		06
28	763				
29	764				
30	765		179764		06
31	766				
32	767				
33	768				
34	769				
35	770				
36	771				
37	772				
38	773				
39	774				
40	775				

W = AIRPLANE GROSS WEIGHT, LBS.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8 OR 13.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-6.5
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date__________

69	70 71	73	77	80
1 5	0 0 2		1 6 P A	
R R	CASE		PROG	

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E	QUAN	LOC	± VALUE	± E
SIG				SIG			
Segm 1	1713			Segm 21	1733	1943	100
2	1714	1985	100	22	1734		
3	1715			23	1735		
4	1716			24	1736	1935	100
5	1717	1959	100	25	1737		
6	1718	1908		26	1738		
7	1719	1798		27	1739	1978	100
8	1720	1629		28	1740		
9	1721	1448		29	1741		
10	1722	134	100	30	1742	1978	100
11	1723			31	1743		
12	1724			32	1744		
13	1725	1304	100	33	1745		
14	1726			34	1746		
15	1727			35	1747		
16	1728	1336	100	36	1748		
17	1729	1448		37	1749		
18	1730	1629		38	1750		
19	1731	1798		39	1751		
20	1732	1888	100	40	1752		

SIG = AIR DENSITY RATIO = ρ/ρ_0.
A VALUE MUST BE ENTERED ONLY FOR THOSE SEGMENTS WITH M3 = 13.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-6.6
STANDARD DATA INPUT 1

Page____ of ______
Prepared by ______
Date __________

RR (69–70)	CASE (71–73)	PROG (77–80)
15	002	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
L or l_t					
Segm 1	3638				
2	3639		500		03
3	3640				
4	3641				
5	3642		1750		04
6	3643		2500		
7	3644				
8	3645				
9	3646				
10	3647		2500		04
11	3648				
12	3649				
13	3650		2500		04
14	3651				
15	3652				
16	3653		2500		04
17	3654				
18	3655				
19	3656				
20	3657		2500		04

QUAN	LOC	±	VALUE	±	E
L or l_t					
Segm 21	3658		1920		04
22	3659				
23	3660				
24	3661		2050		04
25	3662				
26	3663				
27	3664		850		03
28	3665				
29	3666				
30	3667		850		03
31	3668				
32	3669				
33	3670				
34	3671				
35	3672				
36	3673				
37	3674				
38	3675				
39	3676				
40	3677				

L = SCALE OF TURBULENCE, FT.
A VALUE MUST BE ENTERED ONLY FOR THOSE SEGMENTS WITH M3 = 13.

l_t = DISTANCE FROM AIRPLANE C.G. TO LIFT CENTER OF VERTICAL SURFACE, INCHES.
A VALUE MUST BE ENTERED ONLY FOR THOSE SEGMENTS WITH M3 = 14 or 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-7.1
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date ______

R R	CASE	PROG
15	002	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
P_1			
Segm 1	776	1	1
2	777	11	101
3	778	1:	101
4	779	11	101
5	780	142	100
6	781	13	1
7	782	115	1
8	783	1062	1
9	784	1025	1
10	785	10135	100
11	786	11	101
12	787	11	101
13	788	10105	100
14	789	11	101
15	790	11	101
16	791	1013	100
17	792	1025	1
18	793	1062	1
19	794	115	1
20	795	129	100

QUAN	LOC	± VALUE	± E
P_1			
Segm 21	796	14	100
22	797	11	101
23	798	11	101
24	799	139	100
25	800	11	101
26	801	11	101
27	802	17	100
28	803	11	101
29	804	1	1
30	805	1	1
31	806	1	1
32	807	11	101
33	808	1	1
34	809	1	1
35	810	1	1
36	811	1	1
37	812	1	1
38	813	1	1
39	814	1	1
40	815	1	1

P_1 = PORTION OF FLIGHT TIME (OR DISTANCE) SPENT IN NON-STORM TURBULENCE. THE VALUE OF P IN THE FIRST TERM OF GUST EQUATION SPECTRUM INPUT. A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8, 9 OR 13 TO 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-7.2
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date__________

69 70 (R R)	71 73 (CASE)	77 80 (PROG)
15	002	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
P_2			
Segm 1	1473		
2	1474	005	00
3	1475	1	01
4	1476	1	01
5	1477	0033	00
6	1478	002	
7	1479	00095	
8	1480	00028	
9	1481	00011	
10	1482	000095	00
11	1483	1	01
12	1484	1	01
13	1485	000095	00
14	1486	1	01
15	1487	1	01
16	1488	000095	00
17	1489	00011	
18	1490	00028	
19	1491	00095	
20	1492	0019	00

QUAN	LOC	± VALUE	± E
P_2			
Segm 21	1493	0031	00
22	1494	1	01
23	1495	1	01
24	1496	0029	00
25	1497	1	01
26	1498	1	01
27	1499	0045	00
28	1500	1	01
29	1501	1	01
30	1502	005	00
31	1503	1	01
32	1504	1	01
33	1505		
34	1506		
35	1507		
36	1508		
37	1509		
38	1510		
39	1511		
40	1512		

P_2 = PORTION OF FLIGHT TIME (OR DISTANCE) SPENT IN STORM TURBULENCE. THE VALUE OF P IN THE SECOND TERM OF GUST EQUATION SPECTRUM INPUT. ENTER P_2 = 0 IF SECOND TERM IS NOT USED.

A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8, 9 OR 13 TO 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-8.1
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date____________

69 70 R R	71 73 CASE	77 80 PROG
15	002	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E
b_1			
Segm 1	1513		
2	1514	25	01
3	1515		
4	1516	06157	00
5	1517	293	01
6	1518	323	01
7	1519	32	01
8	1520	259	01
9	1521	211	01
10	1522	175	01
11	1523		
12	1524	06157	00
13	1525	16	01
14	1526		
15	1527	0408	00
16	1528	175	01
17	1529	211	01
18	1530	259	01
19	1531	32	01
20	1532	325	01

QUAN	LOC	± VALUE	± E
b_1			
Segm 21	1533	295	01
22	1534		
23	1535	05509	00
24	1536	305	01
25	1537		
26	1538	05509	00
27	1539	265	01
28	1540		
29	1541	05509	00
30	1542	25	01
31	1543		
32	1544	05509	00
33	1545		
34	1546		
35	1547		
36	1548		
37	1549		
38	1550		
39	1551		
40	1552		

b_1 = SCALE PARAMETER IN PROBABILITY DISTRIBUTION OF RMS GUST VELOCITY FOR NON-STORM TURBULENCE. THE VALUE OF b IN THE FIRST TERM OF GUST EQUATION SPECTRUM INPUT.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8, 9 OR 13 TO 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-8.2
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______________

69-70 RR	71-73 CASE	77-80 PROG
15	002	16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
b_2					
Segm 1	1553				
2	1554		5		01
3	1555				
4	1556		2942		00
5	1557		575		01
6	1558		77		01
7	1559		825		01
8	1560		833		01
9	1561		794		01
10	1562		6		01
11	1563		57		01
12	1564		2942		00
13	1565		54		01
14	1566				
15	1567		1687		00
16	1568		6		01
17	1569		794		01
18	1570		833		01
19	1571		825		01
20	1572		775		01
Segm 21	1573		6		01
22	1574				
23	1575		19054		00
24	1576		615		01
25	1577				
26	1578		19054		00
27	1579		51		01
28	1580				
29	1581		19054		00
30	1582		5		01
31	1583				
32	1584		19054		00
33	1585				
34	1586				
35	1587				
36	1588				
37	1589				
38	1590				
39	1591				
40	1592				

b_2 = SCALE PARAMETER IN PROBABILITY DISTRIBUTION OF RMS GUST VELOCITY FOR STORM TURBULENCE. THE VALUE OF b IN THE SECOND TERM OF GUST EQUATION SPECTRUM INPUT. ENTER $b_2 = b_1$ IF SECOND TERM IS NOT USED.

A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8, 9 OR 13 TO 15.

A6PD INPUT DATA

```
BS1
2 2 0 1 48 43 26 3 3 50 1 0 0 20 0 0
90000 -90000 1.0 90000
1112 1416
34 34
-8000 39000
2000 44000
-1.5 1.0
-10 4000 0 4000 10 4000
2528
1 2 2 3 4 4 4 4 4 4 4 5 6 6 7 8 8 8 8 8 0 8 9 10 10 11 12 12 13 14 14 15 16 17
1 2 2 2 3 3 3 3 3 3 3 3 4 4 4 5 5 5 5 5 0 5 5 6 6 6 7 7 7 8 8 8 9 10
1 2 2 3 4 4 4 4 4 4 4 5 6 6 7 8 8 8 8 8 0 8 9 10 10 11 12 12 13 0 0 0 14 15
1 2 2 2 3 3 3 3 3 3 3 3 4 4 4 5 5 5 5 5 0 5 5 6 6 6 7 7 7 0 0 0 8 9
//
```

BASELINE SPECTRUM 1

RR	CASE	FLT	FH	FLTS	LANDGS	FH/FLT	FH/LANDG	LANDGS/FLT
15	1	1A	1099.8	1112	1112	.989	.989	1
15	2	1B	1400.4	1416	1416	.989	.989	1
TOTAL			2500.2	2528	2528	.989	.989	1

REFERENCE RUN NO. 15 CASE NO. 1

THE FOLLOWING DATA IS INPUT DATA

IEND	KEND	I4	S-ULT.	---S---	C-BAR-	---NEND---	L1	IW1	IW2	IW3	IW4	IW5
34	0	0	1.000	2107.20	17.87	0.	1	2	2	1	2	1

SEG.	M3	M5	ISTRES	IA	NIFLAG	------P------	N6	N2	-----AM------	--------F--------	DELTA Y1	DELTA Y11
1	2	2	0	1	2	0.	0	0	1.	-8736.0000	0.0000	0.0000
2	13	7	0	1	2	413.	0	0	1.	8043.0000	.1000	.1000
3	9	7	0	1	2	413.	0	0	1.	8043.0000	.1000	.1000
4	9	7	0	1	1	413.	0	0	1.	8043.0000	.1000	.1000
5	13	7	0	1	2	-1716.	0	0	1.	8415.0000	.1000	.1000
6	13	7	0	1	2	-1716.	0	0	1.	8415.0000	.1000	.1000
7	13	7	0	1	2	-1716.	0	0	1.	8415.0000	.1000	.1000
8	13	7	0	1	2	-1716.	0	0	1.	8415.0000	.1000	.1000
9	13	7	0	1	2	-1716.	0	0	1.	8415.0000	.1000	.1000
10	13	7	0	1	2	-1716.	0	0	1.	8415.0000	.1000	.1000
11	9	7	0	1	2	-1716.	0	0	1.	8415.0000	.1000	.1000
12	9	7	0	1	1	-1716.	0	0	1.	8415.0000	.1000	.1000
13	13	7	0	1	2	-2021.	0	0	1.	8828.0000	.1000	.1000
14	9	7	0	1	2	-2021.	0	0	1.	8828.0000	.1000	.1000
15	9	7	0	1	1	-2021.	0	0	1.	8828.0000	.1000	.1000
16	13	7	0	1	2	-1625.	0	0	1.	8927.0000	.1000	.1000
17	13	7	0	1	2	-1625.	0	0	1.	8927.0000	.1000	.1000
18	13	7	0	1	2	-1625.	0	0	1.	8927.0000	.1000	.1000
19	13	7	0	1	2	-1625.	0	0	1.	8927.0000	.1000	.1000
20	13	7	0	1	2	-1625.	0	0	1.	8927.0000	.1000	.1000
21	13	7	0	1	2	-1625.	0	0	1.	8927.0000	.1000	.1000
22	9	7	0	1	2	-1625.	0	0	1.	8927.0000	.1000	.1000
23	9	7	0	1	1	-1625.	0	0	1.	8927.0000	.1000	.1000
24	13	7	0	1	2	239.	0	0	1.	8877.0000	.1000	.1000
25	9	7	0	1	2	239.	0	0	1.	8877.0000	.1000	.1000
26	9	7	0	1	1	239.	0	0	1.	8877.0000	.1000	.1000
27	13	7	0	1	2	322.	0	0	1.	8877.0000	.1000	.1000
28	9	7	0	1	2	322.	0	0	1.	8877.0000	.1000	.1000
29	9	7	0	1	1	322.	0	0	1.	8877.0000	.1000	.1000
30	13	7	0	1	2	-256.	0	0	1.	8877.0000	.1000	.1000
31	9	7	0	1	2	-256.	0	0	1.	8877.0000	.1000	.1000
32	9	7	0	1	1	-256.	0	0	1.	8877.0000	.1000	.1000
33	1	1	0	1	3	-3805.	0	0	1.	8910.0000	0.0000	0.0000
34	2	2	0	1	2	0.	0	0	1.	-6765.0000	0.0000	0.0000

REFERENCE RUN NO. 15 CASE NO. 1

THE FOLLOWING DATA IS INPUT DATA

SEG.	---N SUB 01---	---N SUB 02---	---N SUB 03---	--SIG DY1--	--SIG DY2--	--SIG DY3--
1	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
2	11.5000	17.2500	0.0000	0.0000	0.0000	0.0000
3	270000.0000	250.0000	0.0000	.1200	.1614	.1000
4	90000.0000	50.0000	0.0000	0.0000	0.0000	0.0000
5	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
6	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
7	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
8	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
9	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
10	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
11	270000.0000	250.0000	0.0000	.1200	.1614	.1000
12	90000.0000	50.0000	0.0000	0.0000	0.0000	0.0000
13	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
14	40000.0000	1000.0000	0.0000	.1452	.1885	.1000
15	21000.0000	62.0000	0.0000	0.0000	0.0000	0.0000
16	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
17	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
18	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
19	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
20	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
21	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
22	190000.0000	0.0000	0.0000	.1336	.1827	.1000
23	170000.0000	450.0000	0.0000	0.0000	0.0000	0.0000
24	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
25	190000.0000	0.0000	0.0000	.1336	.1827	.1000
26	170000.0000	450.0000	0.0000	0.0000	0.0000	0.0000
27	18.0600	18.9800	0.0000	0.0000	0.0000	0.0000
28	190000.0000	0.0000	0.0000	.1336	.1827	.1000
29	170000.0000	450.0000	0.0000	0.0000	0.0000	0.0000
30	11.5000	17.2500	0.0000	0.0000	0.0000	0.0000
31	190000.0000	0.0000	0.0000	133600.0000	.1827	.1000
32	170000.0000	450.0000	0.0000	0.0000	0.0000	0.0000
33	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
34	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000

REFERENCE RUN NO. 15 CASE NO. 1

THE FOLLOWING DATA IS INPUT DATA

SEG.	KSIGMA	SLOPE	VE	-----W----	P1	P2	R1	R2	A-BAR	-------T-------	N
1	0.00000	0.00	0.00	0.	0.000000	0.000000	0.000	0.000	0.000000	1.112	8
2	0.00000	7.15	115.00	166021.	1.000000	.005000	2.500	5.000	0.000000	1544.700	10
3	0.00000	0.00	0.00	0.	1.000000	1.000000	.049	.165	1.000000	.013	12
4	0.00000	0.00	0.00	0.	1.000000	1.000000	.087	.288	1.000000	.013	14
5	0.00000	6.42	281.50	163109.	.420000	.003300	2.930	5.750	0.000000	2236.100	10
6	0.00000	6.43	278.10	163109.	.300000	.002000	3.230	7.700	0.000000	2923.500	10
7	0.00000	6.48	269.40	163109.	.150000	.000950	3.200	8.250	0.000000	6704.300	10
8	0.00000	6.62	259.40	163109.	.062000	.000280	2.590	8.330	0.000000	17824.300	10
9	0.00000	6.87	237.30	163109.	.025000	.000110	2.110	7.940	0.000000	27209.600	10
10	0.00000	7.32	216.00	163109.	.012000	.000095	1.700	5.700	0.000000	50484.800	10
11	0.00000	0.00	0.00	0.	1.000000	1.000000	.049	.165	1.000000	.305	12
12	0.00000	0.00	0.00	0.	1.000000	1.000000	.087	.288	1.000000	.305	14
13	0.00000	7.70	205.00	158677.	.008500	.000098	1.550	4.900	0.000000	196080.000	10
14	0.00000	0.00	0.00	0.	1.000000	1.000000	.052	.099	1.000000	.498	10
15	0.00000	0.00	0.00	0.	1.000000	1.000000	.092	.262	1.000000	.498	18
16	0.00000	7.70	225.00	156213.	.012000	.000095	1.680	5.650	0.000000	19418.000	10
17	0.00000	7.07	250.00	156213.	.025000	.000110	2.110	7.940	0.000000	26599.000	10
18	0.00000	6.56	250.00	156213.	.062000	.000280	2.590	8.330	0.000000	24197.000	10
19	0.00000	6.41	250.00	156213.	.150000	.000950	3.200	8.250	0.000000	11520.000	10
20	0.00000	6.35	250.00	156213.	.290000	.001900	3.250	7.750	0.000000	4428.000	10
21	0.00000	6.35	250.00	156213.	.400000	.003100	2.950	6.000	0.000000	0.000	10
22	0.00000	0.00	0.00	0.	1.000000	1.000000	.054	.054	1.000000	.255	8
23	0.00000	0.00	0.00	0.	1.000000	1.000000	.094	.231	1.000000	.255	24
24	0.00000	6.35	165.00	156213.	.390000	.002900	3.050	6.150	0.000000	2847.000	10
25	0.00000	0.00	0.00	0.	1.000000	1.000000	.054	.054	1.000000	.017	8
26	0.00000	0.00	0.00	0.	1.000000	1.000000	.094	.231	1.000000	.017	24
27	0.00000	6.57	135.00	156213.	.620000	.004200	2.750	5.200	0.000000	1678.100	10
28	0.00000	0.00	0.00	0.	1.000000	1.000000	.054	.054	1.000000	.012	8
29	0.00000	0.00	0.00	0.	1.000000	1.000000	.094	.231	1.000000	.012	24
30	0.00000	6.37	105.00	156213.	1.000000	.005000	2.500	5.000	0.000000	1168.600	10
31	0.00000	0.00	0.00	0.	1.000000	1.000000	.054	.054	1.000000	.011	8
32	0.00000	0.00	0.00	0.	1.000000	1.000000	.094	.231	1.000000	.011	24
33	0.00000	0.00	0.00	0.	0.000000	0.000000	0.000	0.000	0.000000	1.112	22
34	0.00000	0.00	0.00	0.	0.000000	0.000000	0.000	0.000	0.000000	1.112	8

REFERENCE RUN NO. 15 CASE NO. 1

THE FOLLOWING DATA IS INPUT DATA

SEG.	AIR DENSITY RATIO	SCALE OF TURBULENCE
1	0.00000	0.000
2	.98500	500.000
3	0.00000	0.000
4	0.00000	0.000
5	.95900	1750.000
6	.90800	2500.000
7	.79800	2500.000
8	.62900	2500.000
9	.44800	2500.000
10	.32600	2500.000
11	0.00000	0.000
12	0.00000	0.000
13	.27100	2500.000
14	0.00000	0.000
15	0.00000	0.000
16	.32200	2500.000
17	.44800	2500.000
18	.62900	2500.000
19	.79800	2500.000
20	.88800	2500.000
21	.94300	1920.000
22	0.00000	0.000
23	0.00000	0.000
24	.93500	2050.000
25	0.00000	0.000
26	0.00000	0.000
27	.97100	1150.000
28	0.00000	0.000
29	0.00000	0.000
30	.99300	500.000
31	0.00000	0.000
32	0.00000	0.000
33	0.00000	0.000
34	0.00000	0.000

REFERENCE RUN NO. 15 CASE NO. 1

THE FOLLOWING DATA IS INPUT DATA

LL	DELTA Y--TABLE 1	DELTA Y--TABLE 2	DELTA Y--TABLE 3	DELTA Y--TABLE 4	DELTA Y--TABLE 5	DELTA Y--TABLE 6
1	.100	.300	0.000	0.000	0.000	0.000
2	.200	.400	0.000	0.000	0.000	0.000
3	.300	.500	0.000	0.000	0.000	0.000
4	.400	.600	0.000	0.000	0.000	0.000
5	.500	.700	0.000	0.000	0.000	0.000
6	.600	.800	0.000	0.000	0.000	0.000
7	.700	.900	0.000	0.000	0.000	0.000
8	.800	1.000	0.000	0.000	0.000	0.000
9	.900	0.000	0.000	0.000	0.000	0.000
10	1.000	0.000	0.000	0.000	0.000	0.000
11	1.200	0.000	0.000	0.000	0.000	0.000
12	1.400	0.000	0.000	0.000	0.000	0.000
13	1.600	0.000	0.000	0.000	0.000	0.000
14	1.800	0.000	0.000	0.000	0.000	0.000
15	2.000	0.000	0.000	0.000	0.000	0.000
16	2.200	0.000	0.000	0.000	0.000	0.000
17	2.400	0.000	0.000	0.000	0.000	0.000
18	2.600	0.000	0.000	0.000	0.000	0.000
19	2.800	0.000	0.000	0.000	0.000	0.000
20	3.000	0.000	0.000	0.000	0.000	0.000
21	3.200	0.000	0.000	0.000	0.000	0.000
22	3.400	0.000	0.000	0.000	0.000	0.000
23	0.000	0.000	0.000	0.000	0.000	0.000
24	0.000	0.000	0.000	0.000	0.000	0.000
25	0.000	0.000	0.000	0.000	0.000	0.000

REFERENCE RUN NO. 15 CASE NO. 1

THE FOLLOWING DATA IS INPUT DATA

LL	CUM CYCLES TBL 1	CUM CYCLES TBL 2	CUM CYCLES TBL 3	CUM CYCLES TBL 4	CUM CYCLES TBL 5	CUM CYCLES TBL 6
1	1000.000	5000.000	0.000	0.000	0.000	0.000
2	900.000	5000.000	0.000	0.000	0.000	0.000
3	600.000	750.000	0.000	0.000	0.000	0.000
4	450.000	100.000	0.000	0.000	0.000	0.000
5	320.000	15.000	0.000	0.000	0.000	0.000
6	250.000	2.000	0.000	0.000	0.000	0.000
7	210.000	.300	0.000	0.000	0.000	0.000
8	163.000	.050	0.000	0.000	0.000	0.000
9	133.000	0.000	0.000	0.000	0.000	0.000
10	110.000	0.000	0.000	0.000	0.000	0.000
11	70.000	0.000	0.000	0.000	0.000	0.000
12	50.000	0.000	0.000	0.000	0.000	0.000
13	37.000	0.000	0.000	0.000	0.000	0.000
14	26.000	0.000	0.000	0.000	0.000	0.000
15	20.000	0.000	0.000	0.000	0.000	0.000
16	15.500	0.000	0.000	0.000	0.000	0.000
17	11.300	0.000	0.000	0.000	0.000	0.000
18	7.400	0.000	0.000	0.000	0.000	0.000
19	4.800	0.000	0.000	0.000	0.000	0.000
20	3.200	0.000	0.000	0.000	0.000	0.000
21	2.200	0.000	0.000	0.000	0.000	0.000
22	0.000	0.000	0.000	0.000	0.000	0.000
23	0.000	0.000	0.000	0.000	0.000	0.000
24	0.000	0.000	0.000	0.000	0.000	0.000
25	0.000	0.000	0.000	0.000	0.000	0.000

REFERENCE RUN NO. 15 CASE NO. 1

THE FOLLOWING DATA IS INPUT DATA

S-N TABLE = 1 IA AND/OR I4 = 1
NO. OF Y ENTRIES = 3. NO. OF X ENTRIES = 4. MAX CYCLES TO FAILURE = 10000000.

Y	X	Y1,X	Y2,X	Y3,X	Y4,X	Y5,X	Y6,X	Y7,X
-100.000	1000000.0000	0.	0.	0.	0.	0.	0.	0.
0.000	1100000.0000	0.	0.	0.	0.	0.	0.	0.
1.000	1200000.0000	0.	0.	0.	0.	0.	0.	0.
0.000	1300000.0000	0.	0.	0.	0.	0.	0.	0.
0.000	0.0000	0.	0.	0.	0.	0.	0.	0.
0.000	0.0000	0.	0.	0.	0.	0.	0.	0.
0.000	0.0000	0.	0.	0.	0.	0.	0.	0.
0.000	0.0000	0.	0.	0.	0.	0.	0.	0.
0.000	0.0000	0.	0.	0.	0.	0.	0.	0.
0.000	0.0000	0.	0.	0.	0.	0.	0.	0.
0.000	0.0000	0.	0.	0.	0.	0.	0.	0.
0.000	0.0000	0.	0.	0.	0.	0.	0.	0.
0.000	0.0000	0.	0.	0.	0.	0.	0.	0.
0.000	0.0000	0.	0.	0.	0.	0.	0.	0.
0.000	0.0000	0.	0.	0.	0.	0.	0.	0.

Y8,X	Y9,X	Y10,X	Y11,X	Y12,X	Y13,X	Y14,X	Y15,X
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT = 1

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.300	5560.0000	-5678.	-11794.	0.0000
2	.400	5560.0000	-4805.	-12667.	4726.0000
3	.500	834.0000	-3931.	-13541.	722.8000
4	.600	111.2000	-3058.	-14414.	94.5200
5	.700	16.6800	-2184.	-15288.	14.4560
6	.800	2.2240	-1310.	-16162.	1.8904
7	.900	.3336	-437.	-17035.	.2780
8	1.000	.0556			

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT = 2

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	776.2851	9662.	7250.	739.0934
2	.200	37.1917	10467.	6445.	34.7072
3	.300	2.4845	11271.	5641.	2.1916
4	.400	.2929	12075.	4837.	.2419
5	.500	.0510	12880.	4032.	.0409
6	.600	.0101	13684.	3228.	.0080
7	.700	.0021	14488.	2424.	.0016
8	.800	.0004	15293.	1619.	.0003
9	.900	.0001	16097.	815.	.0001
10	1.000	.0000			

GUST ALLEVIATION FACTOR = .602826 A-BAR = .012633

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT = 3

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	473.6701	9662.	7250.	410.8655
2	.200	62.8046	10467.	6445.	54.1676
3	.300	8.6370	11271.	5641.	7.2826
4	.400	1.3545	12075.	4837.	1.0556
5	.500	.2989	12880.	4032.	.1936
6	.600	.1053	13684.	3228.	.0555
7	.700	.0498	14488.	2424.	.0237
8	.800	.0262	15293.	1619.	.0120
9	.900	.0141	16097.	815.	.0064
10	1.000	.0077	16901.	11.	.0035
11	1.100	.0042	17705.	-793.	.0019
12	1.200	.0023			

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT = 4

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	380.3410	9662.	8456.	259.8174
2	.200	120.5237	10467.	8456.	82.2602
3	.300	38.2635	11271.	8456.	26.0652
4	.400	12.1983	12075.	8456.	8.2740
5	.500	3.9243	12880.	8456.	2.6369
6	.600	1.2874	13684.	8456.	.8477
7	.700	.4397	14488.	8456.	.2777
8	.800	.1619	15293.	8456.	.0946
9	.900	.0674	16097.	8456.	.0347
10	1.000	.0327	16901.	8456.	.0143
11	1.100	.0184	17705.	8456.	.0069
12	1.200	.0115	18510.	8456.	.0038
13	1.300	.0077	19314.	8456.	.0024
14	1.400	.0053			

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT = 5

J	DELTA Y	CUMULATIVE CYCLES	SPECTRUM MAX STRESS	MIN STRESS	CYCLES
1	.100	3134.9658	7961.	5437.	2552.4303
2	.200	582.5355	8803.	4595.	471.0385
3	.300	111.4970	9644.	3754.	88.8270
4	.400	22.6699	10486.	2912.	17.5328
5	.500	5.1372	11327.	2071.	3.7758
6	.600	1.3614	12169.	1229.	.9345
7	.700	.4269	13010.	388.	.2741
8	.800	.1529	13852.	-454.	.0935
9	.900	.0593	14693.	-1295.	.0353
10	1.000	.0240			

GUST ALLEVIATION FACTOR = .426416 A-BAR = .019992

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT = 6

			SPECTRUM		
J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	2633.6809	7961.	5437.	2188.6185
2	.200	445.0624	8803.	4595.	365.1473
3	.300	79.9151	9644.	3754.	63.4454
4	.400	16.4697	10486.	2912.	12.1762
5	.500	4.2935	11327.	2071.	2.8419
6	.600	1.4516	12169.	1229.	.8637
7	.700	.5878	13010.	388.	.3279
8	.800	.2599	13852.	-454.	.1409
9	.900	.1190	14693.	-1295.	.0638
10	1.000	.0552			

GUST ALLEVIATION FACTOR = .367915 A-BAR = .017067

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT = 7

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	3189.0936	7961.	5437.	2619.4806
2	.200	569.6130	8803.	4595.	461.2169
3	.300	108.3961	9644.	3754.	84.5244
4	.400	23.8717	10486.	2912.	17.1304
5	.500	6.7413	11327.	2071.	4.2509
6	.600	2.4904	12169.	1229.	1.3895
7	.700	1.1009	13010.	388.	.5716
8	.800	.5293	13852.	-454.	.2664
9	.900	.2628	14693.	-1295.	.1308
10	1.000	.1321			

GUST ALLEVIATION FACTOR = .392309 A-BAR = .017766

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT = 8

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
			SPECTRUM		
1	.100	2702.9021			
			7961.	5437.	2323.4207
2	.200	379.4814			
			8803.	4595.	318.2216
3	.300	61.2599			
			9644.	3754.	47.3369
4	.400	13.9229			
			10486.	2912.	8.9856
5	.500	4.9373			
			11327.	2071.	2.6329
6	.600	2.3044			
			12169.	1229.	1.1181
7	.700	1.1863			
			13010.	388.	.5588
8	.800	.6275			
			13852.	-454.	.2933
9	.900	.3343			
			14693.	-1295.	.1559
10	1.000	.1784			

GUST ALLEVIATION FACTOR = .429446 A-BAR = .019131

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT = 9

J	DELTA Y	CUMULATIVE CYCLES	SPECTRUM MAX STRESS	MIN STRESS	CYCLES
1	.100	1269.6663			
			7961.	5437.	1127.9689
2	.200	141.6973			
			8803.	4595.	119.9798
3	.300	21.7175			
			9644.	3754.	15.4896
4	.400	6.2279			
			10486.	2912.	3.4052
5	.500	2.8227			
			11327.	2071.	1.3452
6	.600	1.4775			
			12169.	1229.	.6802
7	.700	.7974			
			13010.	388.	.3645
8	.800	.4328			
			13852.	-454.	.1976
9	.900	.2352			
			14693.	-1295.	.1074
10	1.000	.1278			

GUST ALLEVIATION FACTOR = .488453 A-BAR = .020657

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =10

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	778.5690			
			7961.	5437.	710.5874
2	.200	67.9817			
			8803.	4595.	56.4724
3	.300	11.5093			
			9644.	3754.	7.6348
4	.400	3.8745			
			10486.	2912.	2.2269
5	.500	1.6476			
			11327.	2071.	.9162
6	.600	.7314			
			12169.	1229.	.4045
7	.700	.3269			
			13010.	388.	.1806
8	.800	.1462			
			13852.	-454.	.0808
9	.900	.0654			
			14693.	-1295.	.0361
10	1.000	.0293			

GUST ALLEVIATION FACTOR = .531838 A-BAR = .021814

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =11

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	10825.8830	7961.	5437.	9390.4635
2	.200	1435.4195	8803.	4595.	1238.0171
3	.300	197.4024	9644.	3754.	166.4459
4	.400	30.9565	10486.	2912.	24.1254
5	.500	6.8311	11327.	2071.	4.4243
6	.600	2.4068	12169.	1229.	1.2680
7	.700	1.1388	13010.	388.	.5411
8	.800	.5977	13852.	-454.	.2751
9	.900	.3226	14693.	-1295.	.1473
10	1.000	.1753	15535.	-2137.	.0799
11	1.100	.0955	16376.	-2978.	.0435
12	1.200	.0520			

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =12

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	8692.8167	7961.	6699.	5938.2097
2	.200	2754.6070	8803.	6699.	1880.0827
3	.300	874.5243	9644.	6699.	595.7292
4	.400	278.7951	10486.	6699.	189.1040
5	.500	89.6911	11327.	6699.	60.2671
6	.600	29.4239	12169.	6699.	19.3755
7	.700	10.0484	13010.	6699.	6.3473
8	.800	3.7011	13852.	6699.	2.1615
9	.900	1.5396	14693.	6699.	.7921
10	1.000	.7475	15535.	6699.	.3271
11	1.100	.4204	16376.	6699.	.1575
12	1.200	.2629	17218.	6699.	.0880
13	1.300	.1750	18059.	6699.	.0548
14	1.400	.1202			

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =13

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	2026.5257	8131.	5483.	1846.7475
2	.200	179.7782	9014.	4600.	146.3281
3	.300	33.4501	9897.	3717.	22.1194
4	.400	11.3307	10780.	2834.	6.7821
5	.500	4.5486	11662.	1952.	2.6683
6	.600	1.8803	12545.	1069.	1.0995
7	.700	.7808	13428.	186.	.4563
8	.800	.3244	14311.	-697.	.1896
9	.900	.1348	15194.	-1580.	.0788
10	1.000	.0560			

GUST ALLEVIATION FACTOR = .552141 A-BAR = .023241

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =14

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	3125.6084	8131.	5483.	2624.8793
2	.200	500.7291	9014.	4600.	412.6478
3	.300	88.0814	9897.	3717.	69.9565
4	.400	18.1249	10780.	2834.	13.5937
5	.500	4.5313	11662.	1952.	3.1896
6	.600	1.3416	12545.	1069.	.8996
7	.700	.4420	13428.	186.	.2883
8	.800	.1537	14311.	-697.	.0989
9	.900	.0548	15194.	-1580.	.0351
10	1.000	.0197			

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =15

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	3553.5437	8131.	6807.	2346.4467
2	.200	1207.0970	9014.	6807.	794.5624
3	.300	412.5346	9897.	6807.	269.8524
4	.400	142.6822	10780.	6807.	92.1894
5	.500	50.4928	11662.	6807.	31.8619
6	.600	18.6309	12545.	6807.	11.2598
7	.700	7.3711	13428.	6807.	4.1445
8	.800	3.2266	14311.	6807.	1.6334
9	.900	1.5932	15194.	6807.	.7113
10	1.000	.8819	16076.	6807.	.3493
11	1.100	.5327	16959.	6807.	.1924
12	1.200	.3402	17842.	6807.	.1158
13	1.300	.2244	18725.	6807.	.0738
14	1.400	.1505	19608.	6807.	.0487
15	1.500	.1019	20490.	6807.	.0326
16	1.600	.0693	21373.	6807.	.0221
17	1.700	.0472	22256.	6807.	.0150
18	1.800	.0322			

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =16

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	379.7669			
			8641.	5963.	340.3249
2	.200	39.4419			
			9534.	5070.	32.8103
3	.300	6.6316			
			10426.	4178.	4.5009
4	.400	2.1307			
			11319.	3285.	1.1948
5	.500	.9359			
			12212.	2392.	.4923
6	.600	.4437			
			13105.	1499.	.2303
7	.700	.2134			
			13997.	607.	.1105
8	.800	.1029			
			14890.	-286.	.0533
9	.900	.0497			
			15783.	-1179.	.0257
10	1.000	.0240			

GUST ALLEVIATION FACTOR = .517557 A-BAR = .024288

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =17

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	1503.2811	8641.	5963.	1304.7865
2	.200	198.4946	9534.	5070.	165.9877
3	.300	32.5069	10426.	4178.	23.8383
4	.400	8.6686	11319.	3285.	4.9244
5	.500	3.7442	12212.	2392.	1.7502
6	.600	1.9940	13105.	1499.	.8709
7	.700	1.1231	13997.	607.	.4825
8	.800	.6406	14890.	-286.	.2742
9	.900	.3664	15783.	-1179.	.1567
10	1.000	.2097			

GUST ALLEVIATION FACTOR = .471520 A-BAR = .022575

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =18

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	3563.4744	8641.	5963.	3076.5299
2	.200	486.9446	9534.	5070.	409.7269
3	.300	77.2176	10426.	4178.	59.6399
4	.400	17.5777	11319.	3285.	11.2811
5	.500	6.2966	12212.	2392.	3.3529
6	.600	2.9437	13105.	1499.	1.4364
7	.700	1.5073	13997.	607.	.7164
8	.800	.7909	14890.	-286.	.3734
9	.900	.4176	15783.	-1179.	.1968
10	1.000	.2208			

GUST ALLEVIATION FACTOR = .424424 A-BAR = .018854

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =19

J	DELTA Y	CUMULATIVE CYCLES	SPECTRUM MAX STRESS	MIN STRESS	CYCLES
1	.100	4928.9348			
			8641.	5963.	4133.1134
2	.200	795.8214			
			9534.	5070.	656.5738
3	.300	139.2475			
			10426.	4178.	109.8860
4	.400	29.3616			
			11319.	3285.	21.0281
5	.500	8.3335			
			12212.	2392.	5.2061
6	.600	3.1274			
			13105.	1499.	1.7523
7	.700	1.3751			
			13997.	607.	.7302
8	.800	.6449			
			14890.	-286.	.3356
9	.900	.3093			
			15783.	-1179.	.1599
10	1.000	.1495			

GUST ALLEVIATION FACTOR = .385778 A-BAR = .016745

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =20

			SPECTRUM		
J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	3357.0728			
			8641.	5963.	2860.2618
2	.200	496.8110			
			9534.	5070.	417.1293
3	.300	79.6817			
			10426.	4178.	64.3025
4	.400	15.3792			
			11319.	3285.	11.3987
5	.500	3.9806			
			12212.	2392.	2.6227
6	.600	1.3578			
			13105.	1499.	.8156
7	.700	.5422			
			13997.	607.	.3112
8	.800	.2310			
			14890.	-286.	.1303
9	.900	.1007			
			15783.	-1179.	.0565
10	1.000	.0442			

GUST ALLEVIATION FACTOR = .366221 A-BAR = .015748

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =21

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	0.0000			
			8641.	5963.	0.0000
2	.200	0.0000			
			9534.	5070.	0.0000
3	.300	0.0000			
			10426.	4178.	0.0000
4	.400	0.0000			
			11319.	3285.	0.0000
5	.500	0.0000			
			12212.	2392.	0.0000
6	.600	0.0000			
			13105.	1499.	0.0000
7	.700	0.0000			
			13997.	607.	0.0000
8	.800	0.0000			
			14890.	-286.	0.0000
9	.900	0.0000			
			15783.	-1179.	0.0000
10	1.000	0.0000			

GUST ALLEVIATION FACTOR = .407111 A-BAR = .017506

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =22

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	7673.2062			
2	.200	1216.6655	8641.	5963.	6456.5407
3	.300	192.9148	9534.	5070.	1023.7507
4	.400	30.5886	10426.	4178.	162.3262
5	.500	4.8501	11319.	3285.	25.7385
6	.600	.7690	12212.	2392.	4.0811
7	.700	.1219	13105.	1499.	.6471
8	.800	.0193	13997.	607.	.1026

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =23

			SPECTRUM		
J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	15051.9960	8641.	7302.	9822.8137
2	.200	5229.1823	9534.	7302.	3405.7376
3	.300	1823.4447	10426.	7302.	1183.2164
4	.400	640.2283	11319.	7302.	412.6175
5	.500	227.6108	12212.	7302.	144.8889
6	.600	82.7219	13105.	7302.	51.5198
7	.700	31.2021	13997.	7302.	18.7302
8	.800	12.4719	14890.	7302.	7.0687
9	.900	5.4032	15783.	7302.	2.8277
10	1.000	2.5756	16675.	7302.	1.2263
11	1.100	1.3493	17568.	7302.	.5852
12	1.200	.7641	18461.	7302.	.3069
13	1.300	.4572	19353.	7302.	.1739
14	1.400	.2833	20246.	7302.	.1041
15	1.500	.1792	21139.	7302.	.0645
16	1.600	.1147	22032.	7302.	.0408
17	1.700	.0738	22924.	7302.	.0261
18	1.800	.0477	23817.	7302.	.0168
19	1.900	.0309	24710.	7302.	.0109
20	2.000	.0200	25602.	7302.	.0070
21	2.100	.0130	26495.	7302.	.0046
22	2.200	.0084	27388.	7302.	.0030
23	2.300	.0055	28281.	7302.	.0019
24	2.400	.0035			

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =24

			SPECTRUM		
J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	1104.4072	10448.	7784.	1038.9995
2	.200	65.4077	11335.	6897.	60.3840
3	.300	5.0237	12223.	6009.	4.3964
4	.400	.6273	13111.	5121.	.5099
5	.500	.1174	13998.	4234.	.0916
6	.600	.0259	14886.	3346.	.0199
7	.700	.0060	15774.	2458.	.0046
8	.800	.0014	16661.	1571.	.0011
9	.900	.0003	17549.	683.	.0002
10	1.000	.0001			

GUST ALLEVIATION FACTOR = .393921 A-BAR = .011180

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =25

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	499.4965	10448.	7784.	420.2962
2	.200	79.2003	11335.	6897.	66.6423
3	.300	12.5580	12223.	6009.	10.5668
4	.400	1.9912	13111.	5121.	1.6755
5	.500	.3157	13998.	4234.	.2657
6	.600	.0501	14886.	3346.	.0421
7	.700	.0079	15774.	2458.	.0067
8	.800	.0013			

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =26

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	979.8276	10448.	9116.	639.4278
2	.200	340.3999	11335.	9116.	221.7006
3	.300	118.6993	12223.	9116.	77.0229
4	.400	41.6764	13111.	9116.	26.8598
5	.500	14.8166	13998.	9116.	9.4317
6	.600	5.3849	14886.	9116.	3.3537
7	.700	2.0311	15774.	9116.	1.2193
8	.800	.8119	16661.	9116.	.4601
9	.900	.3517	17549.	9116.	.1841
10	1.000	.1677	18437.	9116.	.0798
11	1.100	.0878	19325.	9116.	.0381
12	1.200	.0497	20212.	9116.	.0200
13	1.300	.0298	21100.	9116.	.0113
14	1.400	.0184	21988.	9116.	.0068
15	1.500	.0117	22875.	9116.	.0042
16	1.600	.0075	23763.	9116.	.0027
17	1.700	.0048	24651.	9116.	.0017
18	1.800	.0031	25538.	9116.	.0011
19	1.900	.0020	26426.	9116.	.0007
20	2.000	.0013	27314.	9116.	.0005
21	2.100	.0008	28202.	9116.	.0003
22	2.200	.0005	29089.	9116.	.0002
23	2.300	.0004	29977.	9116.	.0001
24	2.400	.0002			

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =27

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	824.9297	10531.	7867.	786.1461
2	.200	38.7835	11418.	6980.	36.4410
3	.300	2.3425	12306.	6092.	2.1125
4	.400	.2300	13194.	5204.	.1957
5	.500	.0343	14081.	4317.	.0282
6	.600	.0061	14969.	3429.	.0050
7	.700	.0011	15857.	2541.	.0009
8	.800	.0002	16744.	1654.	.0002
9	.900	.0000	17632.	766.	.0000
10	1.000	.0000			

GUST ALLEVIATION FACTOR = .479470 A-BAR = .011519

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =28

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	365.4338			
			10531.	7867.	307.4905
2	.200	57.9433			
			11418.	6980.	48.7558
3	.300	9.1875			
			12306.	6092.	7.7207
4	.400	1.4568			
			13194.	5204.	1.2258
5	.500	.2310			
			14081.	4317.	.1944
6	.600	.0366			
			14969.	3429.	.0308
7	.700	.0058			
			15857.	2541.	.0049
8	.800	.0009			

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =29

J	DELTA Y	CUMULATIVE CYCLES	SPECTRUM MAX STRESS	SPECTRUM MIN STRESS	SPECTRUM CYCLES
1	.100	716.8461	10531.	9199.	467.8081
2	.200	249.0380	11418.	9199.	162.1971
3	.300	86.8409	12306.	9199.	56.3503
4	.400	30.4907	13194.	9199.	19.6508
5	.500	10.8399	14081.	9199.	6.9003
6	.600	3.9396	14969.	9199.	2.4536
7	.700	1.4860	15857.	9199.	.8920
8	.800	.5940	16744.	9199.	.3366
9	.900	.2573	17632.	9199.	.1347
10	1.000	.1227	18520.	9199.	.0584
11	1.100	.0643	19408.	9199.	.0279
12	1.200	.0364	20295.	9199.	.0146
13	1.300	.0218	21183.	9199.	.0083
14	1.400	.0135	22071.	9199.	.0050
15	1.500	.0085	22958.	9199.	.0031
16	1.600	.0055	23846.	9199.	.0019
17	1.700	.0035	24734.	9199.	.0012
18	1.800	.0023	25621.	9199.	.0008
19	1.900	.0015	26509.	9199.	.0005
20	2.000	.0010	27397.	9199.	.0003
21	2.100	.0006	28285.	9199.	.0002
22	2.200	.0004	29172.	9199.	.0001
23	2.300	.0003	30060.	9199.	.0001
24	2.400	.0002			

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =30

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	378.9871	9953.	7289.	366.4943
2	.200	12.4928	10840.	6402.	11.7828
3	.300	.7100	11728.	5514.	.6296
4	.400	.0804	12616.	4626.	.0682
5	.500	.0122	13503.	3739.	.0102
6	.600	.0020	14391.	2851.	.0017
7	.700	.0003	15279.	1963.	.0003
8	.800	.0001	16166.	1076.	.0000
9	.900	.0000	17054.	188.	.0000
10	1.000	.0000			

GUST ALLEVIATION FACTOR = .611078 A-BAR = .011071

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =31

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	341.0314	9953.	7289.	286.9574
2	.200	54.0740	10840.	6402.	45.5000
3	.300	8.5740	11728.	5514.	7.2145
4	.400	1.3595	12616.	4626.	1.1439
5	.500	.2156	13503.	3739.	.1814
6	.600	.0342	14391.	2851.	.0288
7	.700	.0054	15279.	1963.	.0046
8	.800	.0009			

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =32

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	668.9776	9953.	8621.	436.5695
2	.200	232.4081	10840.	8621.	151.3661
3	.300	81.0420	11728.	8621.	52.5874
4	.400	28.4546	12616.	8621.	18.3386
5	.500	10.1160	13503.	8621.	6.4395
6	.600	3.6765	14391.	8621.	2.2898
7	.700	1.3868	15279.	8621.	.8325
8	.800	.5543	16166.	8621.	.3142
9	.900	.2401	17054.	8621.	.1257
10	1.000	.1145	17942.	8621.	.0545
11	1.100	.0600	18830.	8621.	.0260
12	1.200	.0340	19717.	8621.	.0136
13	1.300	.0203	20605.	8621.	.0077
14	1.400	.0126	21493.	8621.	.0046
15	1.500	.0080	22380.	8621.	.0029
16	1.600	.0051	23268.	8621.	.0018
17	1.700	.0033	24156.	8621.	.0012
18	1.800	.0021	25043.	8621.	.0007
19	1.900	.0014	25931.	8621.	.0005
20	2.000	.0009	26819.	8621.	.0003
21	2.100	.0006	27707.	8621.	.0002
22	2.200	.0004	28594.	8621.	.0001
23	2.300	.0002	29482.	8621.	.0001
24	2.400	.0002			

GUST ALLEVIATION FACTOR = 0.000000 A-BAR = 1.000000

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =33

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.100	1112.0000	5105.	3768.	111.2000
2	.200	1000.8000	5105.	2877.	333.6000
3	.300	667.2000	5105.	1986.	166.8000
4	.400	500.4000	5105.	1095.	144.5600
5	.500	355.8400	5105.	204.	77.8400
6	.600	278.0000	5105.	-687.	44.4800
7	.700	233.5200	5105.	-1578.	52.2640
8	.800	181.2560	5105.	-2469.	33.3600
9	.900	147.8960	5105.	-3359.	25.5760
10	1.000	122.3200	5105.	-4696.	44.4800
11	1.200	77.8400	5105.	-6478.	22.2400
12	1.400	55.6000	5105.	-8260.	14.4560
13	1.600	41.1440	5105.	-10042.	12.2320
14	1.800	28.9120	5105.	-11824.	6.6720
15	2.000	22.2400	5105.	-13606.	5.0040
16	2.200	17.2360	5105.	-15388.	4.6704
17	2.400	12.5656	5105.	-17170.	4.3368
18	2.600	8.2288	5105.	-18952.	2.8912
19	2.800	5.3376	5105.	-20734.	1.7792
20	3.000	3.5584	5105.	-22516.	1.1120
21	3.200	2.4464	5105.	-24298.	2.4464
22	3.400	0.0000			

REFERENCE RUN NO. 15 CASE NO. 1

SEGMENT =34

SPECTRUM

J	DELTA Y	CUMULATIVE CYCLES	MAX STRESS	MIN STRESS	CYCLES
1	.300	5560.0000			
			-4397.	-9133.	0.0000
2	.400	5560.0000			
			-3721.	-9809.	4726.0000
3	.500	834.0000			
			-3044.	-10486.	722.8000
4	.600	111.2000			
			-2368.	-11162.	94.5200
5	.700	16.6800			
			-1691.	-11839.	14.4560
6	.800	2.2240			
			-1015.	-12515.	1.8904
7	.900	.3336			
			-338.	-13192.	.2780
8	1.000	.0556			

```
SPECTRUM LOADING SEQUENCE GENERATION PROGRAM                    PAGE    1

JOB TITLE   BSI

          NUMBER OF FLIGHT TYPES IS.........................        2
          VALLEY/PEAK COUPLING IS..( 1=AGPA, 2=RANDOM )....        2
          FLIGHT SEQUENCE IS..(0=RANDOM.....................        0
          ...............(N=INPUT FLIGHT SEQUENCE, N SETS)
          FORTRAN UNIT NUMBER FOR LOADING SEQUENCE IS......        1
          NUMBER OF INPUT PEAK LEVELS IS...................       48
          NUMBER OF INPUT RANGE LEVELS IS..................       43
          NUMBER OF INPUT VALLEY/PEAK RATIOS IS............       26
          SAVE FLIGHT SEQUENCE ON MAGNETIC TAPE............        3
          .........................( 0=NO, 3=YES )...
          NUMBER OF POINTS ON THE RANGE VS R INPUT CURVE IS        3
          PRINT FLIGHT SEQUENCE (-1=NO, 0=ALL, N=NUMBER)...       50
          NUMBER  OF SPECTRUM SUMMATIONS TO BE PRINTED.....        1
          PROGRAM TERMINATION FLAG (0=NORMAL, ............
          ........................M=STOP AFTER M FLIGHTS)        0
          ALTERNATE FLIGHT SEQUENCE FLAG...................        0
          .......(0=RANDOM, 1=LO-HI, 2=HI-LO, 3=LO-HI-LO)..
           NUMBER OF HIGHEST PEAKS PER FLIGHT TO BE PRINTED       20
          STARTING VALUE FOR THE GENERATION OF RANDOM......
          ...FLIGHT NUMBERS (0 = DEFAULT DEFINED AS 11111).        0
          STARTING VALUE FOR THE RANDOM CYCLE GENERATION...        0
          .................(0 = DEFAULT DEFINED AS 12345).
          PEAK CLIPPING VALUE IS.....................     90000.
          VALLEY CLIPPING VALUE IS...................    -90000.
          MULTIPLICATION FACTOR IS...................    1.00000
          CYCLE ELIMINATION PEAK VALUE IS............     90000.

INPUT PEAK LEVELS
 -8000.     -7000.     -6000.     -5000.     -4000.     -3000.     -2000.     -1000.         0.      1000.
  2000.      3000.      4000.      5000.      6000.      7000.      8000.      9000.     10000.     11000.
 12000.     13000.     14000.     15000.     16000.     17000.     18000.     19000.     20000.     21000.
 22000.     23000.     24000.     25000.     26000.     27000.     28000.     29000.     30000.     31000.
 32000.     33000.     34000.     35000.     36000.     37000.     38000.     39000.
INPUT RANGE LEVELS
  2000.      3000.      4000.      5000.      6000.      7000.      8000.      9000.     10000.     11000.
 12000.     13000.     14000.     15000.     16000.     17000.     18000.     19000.     20000.     21000.
 22000.     23000.     24000.     25000.     26000.     27000.     28000.     29000.     30000.     31000.
 32000.     33000.     34000.     35000.     36000.     37000.     38000.     39000.     40000.     41000.
 42000.     43000.     44000.
INPUT VALLEY/PEAK RATIOS
 -1.500 -1.400 -1.300 -1.200 -1.100 -1.000  -.900  -.800  -.700  -.600  -.500  -.400  -.300  -.200  -.100   .000   .100   .200
   .300   .400   .500   .600   .700   .800   .900  1.000
INPUT VALLEY/PEAK RATIO VS RANGE CURVE
   -10.000      4000.      0.000      4000.     10.000      4000.
INPUT FLIGHT NUMBERS FOR SPECTRUM SUMMATION PRINT
  2528
```

```
SPECTRUM LOADING SEQUENCE GENERATION PROGRAM                    PAGE    2
JOB TITLE  BS1

FLIGHT      RR      CASE      A6PA SEGMENTS

  1         15        1          1    34
  2         15        2          1    34

FLIGHT TYPE    1 HAS  1112 FLIGHTS AND    34 A6PA SEGMENTS
   F1 SEGMENTS  1  2  2  3  4  4  4  4  4  4  4  5  6  6  7  8  8  8  8  8  0  8  9 10 10 11 12 12 13 14 14 15 16 17
   F2 SEGMENTS  1  2  2  2  3  3  3  3  3  3  3  3  4  4  4  5  5  5  5  5  0  5  5  6  6  6  7  7  7  8  8  8  9 10

FLIGHT TYPE    2 HAS  1416 FLIGHTS AND    34 A6PA SEGMENTS
   F1 SEGMENTS  1  2  2  3  4  4  4  4  4  4  4  5  6  6  7  8  8  8  8  8  0  8  9 10 10 11 12 12 13  0  0  0 14 15
   F2 SEGMENTS  1  2  2  2  3  3  3  3  3  3  3  3  4  4  4  5  5  5  5  5  0  5  5  6  6  6  7  7  7  0  0  0  8  9
```

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 3

JOB TITLE BS1

FLIGHT TYPE 1

F2 SEGMENT	MIN. STRESS	MAX. STRESS	CYCLES	CYCLES/FLIGHT
1	-12667.	-4805.	4726	4.2500
1	-13541.	-3931.	723	.6502
1	-14414.	-3058.	95	.0854
1	-15288.	-2184.	14	.0126
1	-16162.	-1310.	2	.0018
			5560	5.0000
2	7250.	9662.	1150	1.0342
2	6445.	10467.	89	.0800
2	5641.	11271.	9	.0081
2	4837.	12075.	1	.0009
2	8456.	9662.	260	.2338
2	8456.	10467.	82	.0737
2	8456.	11271.	26	.0234
2	8456.	12075.	8	.0072
2	8456.	12880.	3	.0027
2	8456.	13684.	1	.0009
2	8456.	8456.	595	.5351
			2224	2.0000
3	5437.	7961.	20913	18.8067
3	4595.	8803.	3030	2.7248
3	3754.	9644.	474	.4263
3	2912.	10486.	86	.0773
3	2071.	11327.	20	.0180
3	1229.	12169.	7	.0063
3	388.	13010.	3	.0027
3	-454.	13852.	1	.0009
3	-1295.	14693.	1	.0009
3	6699.	7961.	5938	5.3399
3	6699.	8803.	1880	1.6906
3	6699.	9644.	596	.5360
3	6699.	10486.	189	.1700
3	6699.	11327.	60	.0540
3	6699.	12169.	19	.0171
3	6699.	13010.	6	.0054
3	6699.	13852.	2	.0018
3	6699.	14693.	1	.0009
3	6699.	6699.	134	.1205
			33360	30.0000
4	5483.	8131.	4472	4.0216
4	4600.	9014.	559	.5027
4	3717.	9897.	92	.0827
4	2834.	10780.	20	.0180
4	1952.	11662.	6	.0054
4	1069.	12545.	2	.0018
4	186.	13428.	1	.0009
4	6807	8131	2346	2.1097

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 4

JOB TITLE BS1

FLIGHT TYPE 1

F2 SEGMENT	MIN. STRESS	MAX. STRESS	CYCLES	CYCLES/FLIGHT
4	6807.	9014.	795	.7149
4	6807.	9897.	270	.2428
4	6807.	10780.	92	.0827
4	6807.	11662.	32	.0288
4	6807.	12545.	11	.0099
4	6807.	13428.	4	.0036
4	6807.	14311.	2	.0018
4	6807.	15194.	1	.0009
4	6807.	6807.	191	.1718
			8896	8.0000
5	5963.	8641.	18172	16.3417
5	5070.	9534.	2706	2.4335
5	4178.	10426.	424	.3813
5	3285.	11319.	76	.0683
5	2392.	12212.	18	.0162
5	1499.	13105.	6	.0054
5	607.	13997.	2	.0018
5	-286.	14890.	1	.0009
5	-1179.	15783.	1	.0009
5	7302.	8641.	9823	8.8336
5	7302.	9534.	3406	3.0629
5	7302.	10426.	1183	1.0638
5	7302.	11319.	413	.3714
5	7302.	12212.	145	.1304
5	7302.	13105.	52	.0468
5	7302.	13997.	19	.0171
5	7302.	14890.	7	.0063
5	7302.	15783.	3	.0027
5	7302.	16675.	1	.0009
5	7302.	17568.	1	.0009
5	7302.	7302.	237	.2131
			36696	33.0000
6	7784.	10448.	1459	1.3121
6	6897.	11335.	127	.1142
6	6009.	12223.	15	.0135
6	5121.	13111.	2	.0018
6	9116.	10448.	639	.5746
6	9116.	11335.	222	.1996
6	9116.	12223.	77	.0692
6	9116.	13111.	27	.0243
6	9116.	13998.	9	.0081
6	9116.	14886.	3	.0027
6	9116.	15774.	1	.0009
6	9116.	9116.	755	.6790
			3336	3.0000
7	7867.	10531.	1094	.9838

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 5

JOB TITLE BS1

FLIGHT TYPE 1

F2 SEGMENT	MIN. STRESS	MAX. STRESS	CYCLES	CYCLES/FLIGHT
7	6980.	11418.	85	.0764
7	6092.	12306.	10	.0090
7	5204.	13194.	1	.0009
7	9199.	10531.	468	.4209
7	9199.	11418.	162	.1457
7	9199.	12306.	56	.0504
7	9199.	13194.	20	.0180
7	9199.	14081.	7	.0063
7	9199.	14969.	2	.0018
7	9199.	15857.	1	.0009
7	9199.	9199.	318	.2860
			2224	2.0000
8	7289.	9953.	653	.5872
8	6402.	10840.	57	.0513
8	5514.	11728.	8	.0072
8	4626.	12616.	1	.0009
8	8621.	9953.	437	.3930
8	8621.	10840.	151	.1358
8	8621.	11728.	53	.0477
8	8621.	12616.	18	.0162
8	8621.	13503.	6	.0054
8	8621.	14391.	2	.0018
8	8621.	15279.	1	.0009
8	8621.	8621.	837	.7527
			2224	2.0000
9	3768.	5105.	111	.0998
9	2877.	5105.	334	.3004
9	1986.	5105.	167	.1502
9	1095.	5105.	145	.1304
9	204.	5105.	78	.0701
9	-687.	5105.	44	.0396
9	-1578.	5105.	52	.0468
9	-2469.	5105.	33	.0297
9	-3359.	5105.	26	.0234
9	-4696.	5105.	44	.0396
9	-6478.	5105.	22	.0198
9	-8260.	5105.	14	.0126
9	-10042.	5105.	12	.0108
9	-11824.	5105.	7	.0063
9	-13606.	5105.	5	.0045
9	-15388.	5105.	5	.0045
9	-17170.	5105.	4	.0036
9	-18952.	5105.	3	.0027
9	-20734.	5105.	2	.0018
9	-22516.	5105.	1	.0009
9	-24298.	5105.	2	.0018
9	5105.	5105.	1	.0009

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 6

JOB TITLE BS1

FLIGHT TYPE 1

F2 SEGMENT	MIN. STRESS	MAX. STRESS	CYCLES	CYCLES/FLIGHT
10	-9809.	-3721.	4726	4.2500
10	-10486.	-3044.	723	.6502
10	-11162.	-2368.	95	.0854
10	-11839.	-1691.	14	.0126
10	-12515.	-1015.	2	.0018
			5560	5.0000
			101192	91.0000

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 7

JOB TITLE BS1

FLIGHT TYPE 2

F2 SEGMENT	MIN. STRESS	MAX. STRESS	CYCLES	CYCLES/FLIGHT
1	-9544.	-4595.	2832	2.0000
1	-10252.	-3888.	127	.0897
1	-10959.	-3181.	6	.0042
1	-7070.	-7070.	1283	.9061
			4248	3.0000
2	9184.	12810.	111	.0784
2	7975.	14019.	52	.0367
2	6767.	15227.	12	.0085
2	5558.	16436.	2	.0014
2	10997.	12810.	1000	.7062
2	10997.	14019.	197	.1391
2	10997.	15227.	39	.0275
2	10997.	16436.	8	.0056
2	10997.	17644.	2	.0014
2	10997.	10997.	1409	.9951
			2832	2.0000
3	8520.	12188.	17683	12.4880
3	7297.	13411.	3138	2.2161
3	6075.	14633.	564	.3983
3	4852.	15856.	99	.0699
3	3629.	17079.	22	.0155
3	2406.	18302.	7	.0049
3	1184.	19524.	3	.0021
3	-39.	20747.	1	.0007
3	-1262.	21970.	1	.0007
3	10354.	12188.	16848	11.8983
3	10354.	13411.	3323	2.3468
3	10354.	14633.	656	.4633
3	10354.	15856.	131	.0925
3	10354.	17079.	27	.0191
3	10354.	18302.	6	.0042
3	10354.	19524.	2	.0014
3	10354.	20747.	1	.0007
3	10354.	10354.	1384	.9774
			43896	31.0000
4	8347.	12079.	2379	1.6801
4	7103.	13323.	243	.1716
4	5859.	14567.	57	.0403
4	4615.	15811.	17	.0120
4	3370.	17056.	5	.0035
4	2126.	18300.	2	.0014
4	882.	19544.	1	.0007
4	10213.	12079.	3865	2.7295
4	10213.	13323.	338	.2387
4	10213.	14567.	32	.0226
4	10213.	15811.	4	.0028

JOB TITLE BS1

FLIGHT TYPE 2

F2 SEGMENT	MIN. STRESS	MAX. STRESS	CYCLES	CYCLES/FLIGHT
4	10213.	17056.	1	.0007
4	10213.	18300.	1	.0007
4	10213.	10213.	135	.0953
			7080	5.0000
5	3601.	12287.	15246	10.7669
5	7373.	13515.	2473	1.7465
5	6145.	14743.	454	.3206
5	4916.	15972.	80	.0565
5	3688.	17200.	16	.0113
5	2459.	18429.	5	.0035
5	1231.	19657.	2	.0014
5	3.	20885.	1	.0007
5	-1226.	22114.	1	.0007
5	10444.	12287.	15511	10.9541
5	10444.	13515.	2539	1.7931
5	10444.	14743.	421	.2973
5	10444.	15972.	73	.0516
5	10444.	17200.	15	.0106
5	10444.	18429.	4	.0028
5	10444.	19657.	2	.0014
5	10444.	20885.	1	.0007
5	10444.	10444.	1388	.9802
			38232	27.0000
6	10258.	13930.	2259	1.5953
6	9033.	15155.	193	.1363
6	7809.	16379.	34	.0240
6	6585.	17603.	5	.0035
6	5360.	18828.	1	.0007
6	12094.	13930.	2032	1.4350
6	12094.	15155.	333	.2352
6	12094.	16379.	55	.0388
6	12094.	17603.	10	.0071
6	12094.	18828.	2	.0014
6	12094.	20052.	1	.0007
6	12094.	12094.	739	.5219
			5664	4.0000
7	10589.	14261.	2967	2.0953
7	9364.	15486.	220	.1554
7	8140.	16710.	41	.0290
7	6916.	17934.	6	.0042
7	5691.	19159.	1	.0007
7	12425.	14261.	2706	1.9110
7	12425.	15486.	443	.3129
7	12425.	16710.	73	.0516
7	12425.	17934.	13	.0092
7	12425.	19159.	3	.0021

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 9

JOB TITLE BSI

FLIGHT TYPE 2

F2 SEGMENT	MIN. STRESS	MAX. STRESS	CYCLES	CYCLES/FLIGHT
7	12425.	20383.	1	.0007
7	12425.	12425.	606	.4280
			7080	5.0000
8	7143.	8158.	524	.3701
8	6467.	8158.	680	.4802
8	5790.	8158.	58	.0410
8	5114.	8158.	7	.0049
8	4437.	8158.	3	.0021
8	3761.	8158.	1	.0007
8	3084.	8158.	1	.0007
8	2408.	8158.	1	.0007
8	8158.	8158.	141	.0996
			1416	1.0000
9	-7707.	-3711.	2832	2.0000
9	-8278.	-3140.	127	.0897
9	-8849.	-2569.	6	.0042
9	-5709.	-5709.	1283	.9061
			4248	3.0000
			114696	81.0000

```
SPECTRUM LOADING SEQUENCF GENERATION PROGRAM                PACE  -10
JOB TITLE  BS1

FLIGHT NUMBER      1  IS TYPE NUMBER  1     NUMBER OF CYCLES     21   SEQUENCE FOLLOWS
 -12667.20   -4804.80  -12667.20    -4804.80  -14414.40     -4804.80  -12667.20     -4804.80  -12667.20     9662.45
   4595.25    9644.25    4595.25     8802.75    2912.25      7961.25    4595.25      8802.75    4595.25     9014.00
   5482.80    9896.80    4600.00     8641.05    5962.95     10426.45    5070.25      9533.75    5070.25    10530.55
  -1577.50    5105.00   -9809.25    -3720.75   -9809.25     -3720.75   -9809.25     -3720.75   -9809.25    -3720.75
  -9809.25   -3720.75

FLIGHT NUMBER      2  IS TYPE NUMBER  2     NUMBER OF CYCLES     18   SEQUENCE FOLLOWS
  -9544.50   -4595.50   -7070.00    12188.05    8519.95     15856.15   10354.00     15856.15    8519.95    13410.75
   8519.95   13410.75    8519.95    13410.75    8519.95     13410.75    7297.25     12188.05    8519.95    13410.75
   7297.25   12188.05   10213.00    14567.35    8346.85     13515.00    8601.40     14743.40    8601.40    13515.00
   8601.40   13515.00    8601.40    13930.45   10257.55     14261.45

FLIGHT NUMBER      3  IS TYPE NUMBER  2     NUMBER OF CYCLES     25   SEQUENCE FOLLOWS
  -9544.50   -4595.50   -9544.50    12188.05    8519.95     13410.75    7297.25     12188.05    8519.95    14633.45
   8519.95   14633.45    8519.95    13410.75    7297.25     12188.05    7297.25     12188.05    8519.95    13410.75
  10354.00   15856.15    8346.85    13323.25    7373.00     12286.60    8601.40     13515.00    7373.00    12286.60
   7373.00   12286.60    7373.00    12286.60    7373.00     12286.60    8601.40     13515.00    8601.40    13515.00
   8601.40   13515.00    8601.40    13515.00    7373.00     13930.45   10257.55     14261.45   -8278.05    -3710.85

FLIGHT NUMBER      4  IS TYPE NUMBER  1     NUMBER OF CYCLES     19   SEQUENCE FOLLOWS
 -12667.20   -4804.80  -12667.20    -4804.80  -12667.20     -4804.80  -12667.20     -3931.20  -12667.20    11271.05
   4595.25    8802.75    5436.75     9644.25    4595.25      8802.75    4595.25      9014.00    5482.80     9533.75
   5070.25   10426.45    5962.95    10426.45    5962.95     10530.55    1095.50      5105.00  -10485.75    -3720.75
  -9809.25   -3720.75   -9809.25    -3044.25   -9809.25     -3720.75   -9809.25     -3720.75

FLIGHT NUMBER      5  IS TYPE NUMBER  2     NUMBER OF CYCLES     14   SEQUENCE FOLLOWS
  -7070.00   14018.50    8519.95    14633.45    8519.95     13410.75    6074.55     12188.05    7297.25    13410.75
   8519.95   13410.75    8519.95    13410.75    8519.95     13410.75    6144.60     12286.60    8601.40    14743.40
   6144.60   13515.00    8601.40    13515.00    8601.40     13930.45   10257.55     14261.45

FLIGHT NUMBER      6  IS TYPE NUMBER  1     NUMBER OF CYCLES     20   SEQUENCE FOLLOWS
 -12667.20   -4804.80  -13540.80    -4804.80  -12667.20     -4804.80  -12667.20     -4804.80  -12667.20     9662.45
   4595.25    8802.75    5436.75     9644.25    3753.75      9014.00    4600.00      9533.75    5962.95    10426.45
   5962.95   10426.45    5070.25     9533.75    4177.55      9533.75    5962.95     11319.15    3284.85    11335.25
  -9809.25   -3720.75   -9809.25    -3720.75   -9809.25     -3720.75   -9809.25     -3720.75   -9809.25    -3720.75

FLIGHT NUMBER      7  IS TYPE NUMBER  1     NUMBER OF CYCLES     25   SEQUENCE FOLLOWS
 -12667.20   -4804.80  -12667.20    -4804.80  -12667.20     -4804.80  -12667.20     -4804.80  -12667.20     9662.45
   3753.75    8802.75    5436.75     9644.25    4595.25      8802.75    4595.25     10485.75    5436.75     9644.25
   5070.25    9533.75    5962.95    10426.45    5962.95     11319.15    5070.25      9533.75    4177.55     8641.05
   5070.25    9533.75    5962.95    10426.45    5962.95     10426.45    5962.95     10426.45    5962.95    10530.55
  -9809.25   -3044.25   -9809.25    -3720.75   -9809.25     -3720.75   -9809.25     -3720.75  -11838.75    -3720.75

FLIGHT NUMBER      8  IS TYPE NUMBER  1     NUMBER OF CYCLES     16   SEQUENCE FOLLOWS
 -12667.20   -3931.20  -12667.20    -4804.80  -12667.20     -4804.80  -12667.20     -3931.20  -13540.80     9662.45
   4595.25    8802.75    4595.25     9896.80    5482.80      9896.80    5070.25      9533.75    5962.95    12211.85
   5070.25   11335.25   -9809.25    -3720.75   -9809.25     -3720.75   -9809.25     -3720.75   -9809.25    -3720.75
  -9809.25   -3044.25
```

JOB TITLE   BS1

```
FLIGHT NUMBER     9  IS TYPE NUMBER  1    NUMBER OF  CYCLES      20    SEQUENCE  FOLLOWS
 -12667.20   -4804.80   -12667.20    -4804.80   -12667.20    -4804.80  -14414.40    -3931.20   -12667.20    9662.45
   5436.75   10485.75     3753.75     8802.75     4595.25     9644.25    5482.80     9896.80     5482.80    9896.80
   5482.80    9533.75     4177.55     9533.75     4177.55     9533.75    5070.25    10447.55     7784.45   12305.95
  -9809.25   -3720.75    -9809.25    -2367.75    -9809.25    -3720.75   -9809.25    -3720.75   -10485.75   -3720.75

FLIGHT NUMBER    10  IS TYPE NUMBER  2    NUMBER OF  CYCLES      13    SEQUENCE  FOLLOWS
 -10958.50   12809.90     8519.95    13410.75     8519.95    13410.75    7297.25    12188.05     6074.55   12188.05
   5858.65   12286.60     8601.40    13515.00     8601.40    14743.40    8601.40    13515.00     8601.40   13515.00
   8601.40   14261.45    10588.55    15485.75    -7707.15    -3139.95

FLIGHT NUMBER    11  IS TYPE NUMBER  2    NUMBER OF  CYCLES      18    SEQUENCE  FOLLOWS
  -9544.50   -4595.50    -9544.50    12809.90     6074.55    13410.75    4851.85    12188.05     7297.25   12188.05
   8519.95   13410.75     8519.95    13410.75     6074.55    12188.05    8346.85    13323.25     7373.00   13515.00
   8601.40   15971.80     8601.40    13515.00     7373.00    12286.60    8601.40    14743.40     8601.40   13515.00
   8601.40   13930.45    10257.55    14261.45    10588.55    15485.75

FLIGHT NUMBER    12  IS TYPE NUMBER  1    NUMBER OF  CYCLES      14    SEQUENCE  FOLLOWS
 -12667.20   -4804.80   -12667.20    -4804.80   -12667.20    -4804.80  -12667.20    -4804.80   -13540.80    9662.45
   5436.75    9644.25     4600.00     9533.75     5070.25     9533.75    5962.95    10840.25    -9809.25   -3044.25
 -11162.25   -3720.75    -9809.25    -3720.75    -9809.25    -3720.75   -9809.25    -3720.75

FLIGHT NUMBER    13  IS TYPE NUMBER  1    NUMBER OF  CYCLES      15    SEQUENCE  FOLLOWS
 -12667.20   -4804.80   -12667.20    -3931.20   -12667.20    -4804.80  -12667.20    -4804.80   -12667.20    9662.45
   4595.25    9533.75     5070.25    11319.15     5070.25     9533.75    5070.25    10530.55     7289.45   11727.95
 -10485.75   -3720.75    -9809.25    -3720.75    -9809.25    -3720.75   -9809.25    -3720.75    -9809.25   -3044.25

FLIGHT NUMBER    14  IS TYPE NUMBER  2    NUMBER OF  CYCLES      13    SEQUENCE  FOLLOWS
  -9544.50   12809.90     7297.25    13410.75     8519.95    13410.75    8601.40    14743.40     6144.60   12286.60
   8601.40   13515.00     8601.40    13515.00     6144.60    12286.60    7373.00    12286.60     7373.00   13515.00
   8601.40   13930.45    10257.55    14261.45    10588.55    15485.75

FLIGHT NUMBER    15  IS TYPE NUMBER  2    NUMBER OF  CYCLES      16    SEQUENCE  FOLLOWS
 -10251.50   -4595.50    -7070.00    12809.90     8519.95    13410.75    7297.25    12188.05     7297.25   12188.05
   8519.95   13410.75     7297.25    13410.75     8519.95    13410.75    8346.85    13323.25     6144.60   12286.60
   8601.40   13515.00     7373.00    12286.60     8601.40    13515.00    8601.40    13930.45    10257.55   15154.75
  -8278.05   -3710.85

FLIGHT NUMBER    16  IS TYPE NUMBER  2    NUMBER OF  CYCLES      16    SEQUENCE  FOLLOWS
  -9544.50   12809.90     8519.95    13410.75     8519.95    13410.75    8519.95    13410.75     8519.95   13410.75
   7297.25   12188.05     7297.25    12188.05     6074.55    12188.05    7102.75    12286.60     8601.40   13515.00
   8601.40   13515.00     7373.00    12286.60     8601.40    13515.00    8601.40    15971.80     8601.40   13515.00
   8601.40   14261.45

FLIGHT NUMBER    17  IS TYPE NUMBER  1    NUMBER OF  CYCLES      24    SEQUENCE  FOLLOWS
 -12667.20   -4804.80   -13540.80    -3931.20   -12667.20    -4804.80  -12667.20    -3931.20   -12667.20    9662.45
   4595.25    8802.75     5436.75    10485.75     4595.25     8802.75    4595.25     9644.25     6699.00   11327.25
   4600.00    9896.80     4600.00    10779.60     5962.95    12211.85    4177.55     9533.75     5070.25    9533.75
   5070.25   10426.45     5070.25     9533.75     5962.95    10530.55    6401.75    11727.95    -9809.25   -3720.75
```

JOB TITLE   BS1

```
 -9809.25   -3044.25   -9809.25   -3044.25   -9809.25   -3720.75   -9809.25   -3720.75

FLIGHT NUMBER   18   IS TYPE NUMBER   1   NUMBER OF CYCLES   18   SEQUENCE FOLLOWS
 -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20    9662.45
   5436.75   10485.75    5436.75    9644.25    4595.25   10485.75    4595.25    9014.00    4600.00    9533.75
   5070.25    9533.75    5070.25    9533.75    5962.95   10530.55   -9809.25   -3720.75   -9809.25   -3720.75
  -9809.25   -3044.25   -9809.25   -3720.75   -9809.25   -3720.75

FLIGHT NUMBER   19   IS TYPE NUMBER   1   NUMBER OF CYCLES   24   SEQUENCE FOLLOWS
 -12667.20   -4804.80  -14414.40   -4804.80  -13540.80   -4804.80  -12667.20   -4804.80  -12667.20   10466.75
   4595.25    8802.75    4595.25    9644.25    5436.75    9644.25    5482.80    9896.80    6807.00   11662.40
   5482.80   10426.45    2392.15    8641.05    5962.95   10426.45    5070.25    9533.75    5070.25    9533.75
   5070.25   11319.15    5070.25   11319.15    5962.95   10447.55    6009.05   10530.55   -9809.25   -3720.75
 -10485.75   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75

FLIGHT NUMBER   20   IS TYPE NUMBER   1   NUMBER OF CYCLES   20   SEQUENCE FOLLOWS
 -13540.80   -4804.80  -12667.20   -4804.80  -12667.20   -3531.20  -12667.20   -4804.80  -13540.80    9662.45
   4595.25    8802.75    4595.25    8802.75    4595.25    8802.75    5436.75    9644.25    5070.25    9533.75
   5070.25    9533.75    5070.25    9533.75    5962.95   10447.55    9116.00   14886.05   -3359.50    5105.00
 -10485.75   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75

FLIGHT NUMBER   21   IS TYPE NUMBER   2   NUMBER OF CYCLES   16   SEQUENCE FOLLOWS
  -9544.50   -4595.50   -9544.50   10997.00    8519.95   13410.75    8519.95   13410.75    8519.95   13410.75
   7297.25   12188.05   10354.00   14633.45    8519.95   14633.45    4614.55   12286.60    6144.60   12286.60
   7373.00   12286.60    8601.40   13515.00    8601.40   14743.40   10257.55   16379.05   10257.55   14261.45
  12425.00   16710.05

FLIGHT NUMBER   22   IS TYPE NUMBER   1   NUMBER OF CYCLES   18   SEQUENCE FOLLOWS
 -13540.80   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20    9662.45
   5436.75    9644.25    4595.25    8802.75    4595.25    9014.00    5482.80    9533.75    5070.25   10426.45
   5070.25    9533.75    5962.95   11418.25     204.50    5105.00   -9809.25   -3720.75   -9809.25   -3720.75
  -9809.25   -3720.75   -9809.25   -3720.75  -10485.75   -3720.75

FLIGHT NUMBER   23   IS TYPE NUMBER   1   NUMBER OF CYCLES   20   SEQUENCE FOLLOWS
 -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -13540.80   -4804.80  -13540.80    9662.45
   4595.25    8802.75    5436.75    9644.25    5436.75    9644.25    5436.75    9644.25    5436.75    9644.25
   5482.80   10779.60    5070.25    9533.75    5070.25    9533.75    5070.25   11335.25    7784.45   12305.95
  -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75

FLIGHT NUMBER   24   IS TYPE NUMBER   2   NUMBER OF CYCLES   23   SEQUENCE FOLLOWS
  -9544.50   -3888.50   -9544.50   12809.90    8519.95   13410.75    8519.95   13410.75    7297.25   13410.75
   8519.95   13410.75    7297.25   12188.05    8519.95   14633.45    8519.95   13410.75    8346.85   14567.35
   6144.60   12286.60    8601.40   14743.40    8601.40   13515.00    7373.00   12286.60    6144.60   12286.60
   7373.00   12286.60    8601.40   13515.00    7373.00   12286.60    8601.40   13515.00    7373.00   12286.60
   7373.00   13930.45    9033.25   15154.75   -7707.15   -3139.95

FLIGHT NUMBER   25   IS TYPE NUMBER   2   NUMBER OF CYCLES   17   SEQUENCE FOLLOWS
  -9544.50   -4595.50   -9544.50   12188.05    7297.25   12188.05    7297.25   13410.75    8519.95   13410.75
   8519.95   13410.75    8519.95   13410.75    7297.25   12188.05    7297.25   12286.60    7373.00   12286.60
   7373.00   12286.60    8601.40   13515.00    8601.40   13515.00    8601.40   14743.40    8601.40   14261.45
```

JOB TITLE BS1

```
10588.55    15485.75    -7707.15    -3139.95

FLIGHT NUMBER   26   IS TYPE NUMBER  2   NUMBER OF CYCLES   15   SEQUENCE FOLLOWS
 -9544.50    -4595.50    -9544.50    -4595.50    -9544.50    12188.05     8519.95    13410.75     8519.95    13410.75
  7297.25    12188.05     7297.25    13410.75     7297.25    12188.05     4851.85    12188.05     8519.95    14633.45
  7297.25    13410.75     8346.85    13515.00     7373.00    12286.60     7373.00    12286.60     8601.40    15154.75

FLIGHT NUMBER   27   IS TYPE NUMBER  2   NUMBER OF CYCLES   17   SEQUENCE FOLLOWS
 -9544.50    -4595.50    -9544.50    -4595.50    -9544.50    12188.05     7297.25    13410.75     4851.85    12188.05
  7297.25    12188.05     6074.55    13410.75     8346.85    13323.25     7373.00    12286.60     8601.40    13515.00
  6144.60    13515.00     8601.40    13515.00    10257.55    15154.75    10257.55    14261.45    10588.55    16710.05
  9364.25    14261.45    -8278.05    -3710.85

FLIGHT NUMBER   28   IS TYPE NUMBER  2   NUMBER OF CYCLES   16   SEQUENCE FOLLOWS
 -9544.50    -4595.50    -9544.50    -4595.50    -9544.50    12809.90     8519.95    14633.45     7297.25    12188.05
  7297.25    15856.15     8519.95    14633.45     7297.25    12286.60     8601.40    13515.00     8601.40    13515.00
  6144.60    13515.00     7373.00    13515.00     8601.40    13515.00     7373.00    12286.60    10257.55    15154.75
 10257.55    15485.75

FLIGHT NUMBER   29   IS TYPE NUMBER  2   NUMBER OF CYCLES   22   SEQUENCE FOLLOWS
 -9544.50    14018.50     7297.25    12188.05     7297.25    12188.05     8519.95    13410.75     7297.25    12188.05
  8519.95    13410.75     8519.95    14633.45     8519.95    14633.45     8519.95    13410.75     8519.95    14633.45
  7297.25    12188.05     7297.25    12188.05     8519.95    13410.75     8346.85    13323.25     8601.40    13515.00
  7373.00    12286.60     7373.00    13515.00     8601.40    13515.00     8601.40    13515.00     8601.40    13930.45
 10257.55    15154.75    10588.55    15485.75

FLIGHT NUMBER   30   IS TYPE NUMBER  2   NUMBER OF CYCLES   17   SEQUENCE FOLLOWS
-10251.50    -4595.50    -9544.50    12809.90     8519.95    13410.75     7297.25    12188.05     7297.25    12188.05
  8519.95    13410.75     8519.95    14633.45     8519.95    14633.45     8519.95    13410.75     8346.85    13323.25
  8346.85    13515.00     8601.40    13515.00     8601.40    13930.45     9033.25    15485.75    10588.55    15485.75
 10588.55    15485.75    -8278.05    -3710.85

FLIGHT NUMBER   31   IS TYPE NUMBER  2   NUMBER OF CYCLES   19   SEQUENCE FOLLOWS
 -9544.50    -4595.50    -9544.50    12809.90     6074.55    13410.75     7297.25    12188.05     6074.55    12188.05
  8519.95    13410.75     8519.95    13410.75     8519.95    13410.75     8519.95    14633.45     8519.95    14633.45
  8519.95    13410.75     8346.85    13515.00     8601.40    13515.00     8601.40    13515.00     8601.40    13515.00
  8601.40    13515.00    10444.00    14743.40     8601.40    13930.45     9033.25    14261.45

FLIGHT NUMBER   32   IS TYPE NUMBER  2   NUMBER OF CYCLES   13   SEQUENCE FOLLOWS
 -9544.50    -4595.50    -9544.50    14018.50     8519.95    13410.75     8519.95    13410.75     7297.25    12188.05
  8519.95    13410.75     8346.85    13515.00     8601.40    13515.00     8601.40    13515.00     7373.00    12286.60
  8601.40    13930.45    10257.55    15485.75    10588.55    15485.75

FLIGHT NUMBER   33   IS TYPE NUMBER  2   NUMBER OF CYCLES   18   SEQUENCE FOLLOWS
-10251.50    -4595.50    -9544.50    12188.05     8519.95    13410.75     8519.95    13410.75     8519.95    13410.75
  8519.95    13410.75     8519.95    13410.75     7297.25    12188.05     7102.75    12079.15     5858.65    12079.15
  7373.00    12286.60     7373.00    12286.60     7373.00    12286.60     8601.40    13515.00     7373.00    13930.45
 10257.55    16710.05    12425.00    16710.05    10588.55    15485.75
```

JOB TITLE   BS1

```
FLIGHT NUMBER   34  IS TYPE NUMBER  2    NUMBER OF CYCLES      19   SEQUENCE FOLLOWS
   -9544.50   -4595.50    -9544.50   -4595.50   -9544.50   14018.50    8519.95   13410.75    7297.25   12188.05
    6074.55   12188.05     7297.25   12188.05    7297.25   13410.75    6074.55   12188.05    8519.95   13410.75
    8519.95   13410.75     8519.95   13410.75    8519.95   13410.75    8346.85   13323.25    8346.85   14743.40
    7373.00   13515.00     8601.40   13515.00    7373.00   13930.45    9033.25   14261.45

FLIGHT NUMBER   35  IS TYPE NUMBER  2    NUMBER OF CYCLES      19   SEQUENCE FOLLOWS
   -9544.50   -4595.50    -9544.50   12188.05    8519.95   13410.75    7297.25   13410.75    8519.95   14633.45
    7297.25   12188.05     8519.95   13410.75    8346.85   13323.25   10213.00   14567.35    6144.60   12286.60
    8601.40   13515.00     8601.40   14743.40    8601.40   14743.40    7373.00   12286.60    7373.00   13515.00
    6144.60   12286.60    10444.00   14743.40    8601.40   14261.45    9364.25   14261.45

FLIGHT NUMBER   36  IS TYPE NUMBER  2    NUMBER OF CYCLES      16   SEQUENCE FOLLOWS
   -9544.50   -4595.50    -9544.50   12188.05    7297.25   13410.75    7297.25   13410.75    7297.25   12188.05
    8519.95   13410.75     8519.95   13410.75    8519.95   13410.75    6144.60   12286.60    8601.40   13515.00
    7373.00   13515.00     8601.40   13515.00    8601.40   13515.00    9033.25   13930.45   10257.55   16379.05
   10588.55   15485.75

FLIGHT NUMBER   37  IS TYPE NUMBER  2    NUMBER OF CYCLES      20   SEQUENCE FOLLOWS
   -9544.50   15227.10     7297.25   13410.75    7297.25   12188.05    7297.25   12188.05    8519.95   13410.75
    8519.95   13410.75     7297.25   13410.75    8346.85   13323.25    8346.85   13515.00    8601.40   13515.00
    7373.00   13515.00     8601.40   13515.00    8601.40   13515.00    8601.40   13515.00    8601.40   13515.00
    8601.40   13515.00     8601.40   14743.40    9033.25   16379.05    9364.25   14261.45   -8278.05   -3710.85

FLIGHT NUMBER   38  IS TYPE NUMBER  1    NUMBER OF CYCLES      25   SEQUENCE FOLLOWS
  -13540.80   -4804.80   -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   10466.75
    4595.25    8802.75     4595.25    8802.75    5436.75    9644.25    3717.20    9014.00    5482.80   10779.60
    5482.80   10426.45     5962.95   10426.45    4177.55    8641.05    5962.95   13104.55    5962.95   10426.45
    5070.25    9533.75     4177.55    8641.05    5962.95   11319.15    5962.95   10840.25    1095.50    5105.00
   -9809.25   -3044.25    -9809.25   -3720.75   -9809.25   -3044.25   -9809.25   -3720.75   -9809.25   -3720.75

FLIGHT NUMBER   39  IS TYPE NUMBER  2    NUMBER OF CYCLES      19   SEQUENCE FOLLOWS
   -9544.50   -4595.50    -9544.50   -4595.50   -9544.50   12809.90    7297.25   13410.75    7297.25   12188.05
    8519.95   14633.45     7297.25   12188.05    7297.25   13410.75    8519.95   14633.45    8519.95   13410.75
    8519.95   14633.45     8519.95   13410.75    8519.95   13410.75    8346.85   17055.55    8346.85   14743.40
    7373.00   12286.60     8601.40   13515.00    8601.40   13930.45   10257.55   14261.45

FLIGHT NUMBER   40  IS TYPE NUMBER  1    NUMBER OF CYCLES      16   SEQUENCE FOLLOWS
  -12667.20   -4804.80   -12667.20   -3931.20  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20    9662.45
    4595.25    8802.75     4595.25    9014.00    5482.80   10426.45    5962.95   11319.15    5070.25    9533.75
    5962.95   10530.55   -10485.75   -3720.75   -9809.25   -3044.25   -9809.25   -3720.75   -9809.25   -3720.75
   -9809.25   -3720.75

FLIGHT NUMBER   41  IS TYPE NUMBER  1    NUMBER OF CYCLES      19   SEQUENCE FOLLOWS
  -12667.20   -3931.20   -14414.40   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20    9662.45
    4595.25    8802.75     4595.25    9014.00    4600.00   10426.45    5070.25    9533.75    4177.55    9533.75
    5070.25    9533.75     5962.95   10426.45    5962.95   11335.25   -3359.50    5105.00  -10485.75   -3720.75
   -9809.25   -3044.25    -9809.25   -3720.75  -10485.75   -3044.25  -11838.75   -3720.75
```

JOB TITLE   BS1

```
FLIGHT NUMBER   42  IS TYPE NUMBER  1    NUMBER OF CYCLES    18   SEQUENCE FOLLOWS
 -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -14414.40    9662.45
   4595.25    8802.75    4595.25    9644.25    3717.20    9014.00    6807.00   12545.20    5482.80   11319.15
   5962.95   12211.85    4177.55    3641.05    4177.55   10840.25   -9809.25   -3720.75   -9809.25   -3720.75
  -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75

FLIGHT NUMBER   43  IS TYPE NUMBER  2    NUMBER OF CYCLES    18   SEQUENCE FOLLOWS
  -9544.50   -4595.50   -9544.50   15227.10    7297.25   12188.05    8519.95   13410.75    8519.95   13410.75
   8519.95   15856.15    7297.25   13410.75    8346.85   13515.00    7373.00   12286.60    8601.40   13515.00
   7373.00   12286.60    8601.40   13515.00    8601.40   13515.00    8601.40   13515.00    7373.00   12286.60
   8601.40   13930.45   10257.55   14261.45    5691.35   14261.45

FLIGHT NUMBER   44  IS TYPE NUMBER  1    NUMBER OF CYCLES    25   SEQUENCE FOLLOWS
 -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -3931.20  -13540.80    9662.45
   5436.75    9644.25    3753.75    7961.25    4595.25    9644.25    5436.75    9644.25    4595.25    8802.75
   4595.25    8802.75    5436.75    9644.25    4595.25    9896.80    4600.00    9533.75    5070.25   10426.45
   5962.95   10426.45    5070.25    9533.75    5962.95   10426.45    5070.25   11319.15    6896.75   11418.25
  -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3044.25  -10485.75   -3720.75

FLIGHT NUMBER   45  IS TYPE NUMBER  1    NUMBER OF CYCLES    21   SEQUENCE FOLLOWS
 -12667.20   -3931.20  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20    9662.45
   4595.25    8802.75    4595.25    8802.75    4595.25    9014.00    4600.00    9533.75    5070.25    9533.75
   5962.95   13104.55    5070.25    9533.75    5070.25    9533.75    5962.95   10426.45    5962.95   10530.55
   1095.50    5105.00  -10485.75   -3044.25  -10485.75   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75
  -9809.25   -3720.75

FLIGHT NUMBER   46  IS TYPE NUMBER  2    NUMBER OF CYCLES    14   SEQUENCE FOLLOWS
  -9544.50   -4595.50   -9544.50   -4595.50   -9544.50   12809.90    8519.95   14633.45    7297.25   12188.05
   7297.25   13410.75    8519.95   14633.45    8519.95   13410.75    8346.85   13323.25    8346.85   13323.25
   8601.40   13515.00    8601.40   13515.00    8601.40   15485.75   10588.55   15485.75

FLIGHT NUMBER   47  IS TYPE NUMBER  1    NUMBER OF CYCLES    19   SEQUENCE FOLLOWS
 -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -13540.80    9662.45
   3753.75    8802.75    4595.25    9644.25    3753.75    7961.25    5436.75    9644.25    5436.75    9533.75
   5070.25    9533.75    5962.95   12211.85    5070.25    9533.75    5070.25   10840.25   -9809.25   -3720.75
  -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75

FLIGHT NUMBER   48  IS TYPE NUMBER  1    NUMBER OF CYCLES    19   SEQUENCE FOLLOWS
 -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20    9662.45
   4595.25    8802.75    4595.25    8802.75    4595.25    9644.25    4595.25    8802.75    4595.25    9014.00
   5482.80   10426.45    5962.95   10426.45    3284.85    9533.75    5962.95   11335.25   -9809.25   -3720.75
 -10485.75   -3044.25   -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75

FLIGHT NUMBER   49  IS TYPE NUMBER  1    NUMBER OF CYCLES    20   SEQUENCE FOLLOWS
 -12667.20   -4804.80  -13540.80   -4804.80  -12667.20   -4804.80  -12667.20   -4804.80  -12667.20    9662.45
   8456.00   12879.65    4595.25    8802.75    5436.75    9644.25    5436.75    9644.25    5436.75   10779.60
   5482.80   10426.45    5070.25   10426.45    5962.95   11319.15    5962.95   10530.55  -11824.00    5105.00
  -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75   -9809.25   -3720.75
```

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 16

JOB TITLE BS1

FLIGHT NUMBER 50 IS TYPE NUMBER 2 NUMBER OF CYCLES 24 SEQUENCE FOLLOWS

-9544.50	-4595.50	-9544.50	-4595.50	-9544.50	14018.50	7297.25	12188.05	7297.25	12188.05
8519.95	13410.75	8519.95	13410.75	7297.25	12188.05	8519.95	13410.75	7297.25	12188.05
8519.95	13410.75	8519.95	13410.75	7297.25	12188.05	6074.55	13410.75	8346.85	14567.35
8601.40	13515.00	7373.00	14743.40	8601.40	13515.00	8601.40	13515.00	10444.00	15971.80
8601.40	13515.00	7373.CC	12286.60	8601.40	15154.75	10588.55	15485.75		

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 17

JOB TITLE BS1

SPECTRUM SUMMATION FOR A TOTAL OF 2528 FLIGHTS AND 45807 CYCLES

VALLEY/PEAK RATIO RANGE		-1.50		-1.40		-1.30		-1.20		-1.10		-1.00
BELOW OR EQUAL 2000.	0	71	0	174	0	725	0	189	0	25	0	3
2000. 3000.	0	71	0	174	0	725	0	189	0	25	0	3
3000. 4000.	0	71	0	174	0	725	0	189	0	25	0	3
4000. 5000.	0	71	0	174	0	725	0	189	0	25	0	3
5000. 6000.	0	71	0	174	0	725	0	189	0	25	0	3
6000. 7000.	0	71	0	174	0	725	0	189	0	25	0	3
7000. 8000.	0	71	0	174	0	725	0	189	0	25	0	3
8000. 9000.	0	71	0	174	0	725	0	189	0	25	0	3
9000. 10000.	0	71	0	174	0	725	0	189	0	25	0	3
10000. 11000.	0	71	0	174	0	725	0	189	0	25	0	3
11000. 12000.	0	71	0	174	0	725	22	189	0	25	0	3
12000. 13000.	0	71	0	174	0	725	0	167	0	25	0	3
13000. 14000.	14	71	0	174	0	725	0	167	0	25	0	3
14000. 15000.	0	57	0	174	0	725	0	167	0	25	0	3
15000. 16000.	12	57	0	174	0	725	0	167	0	25	0	3
16000. 17000.	7	45	0	174	0	725	0	167	0	25	0	3
17000. 18000.	0	38	0	174	0	725	0	167	0	25	0	3
18000. 19000.	5	38	0	174	0	725	0	167	0	25	0	3
19000. 20000.	0	33	0	174	0	725	0	167	0	25	0	3
20000. 21000.	5	33	0	174	0	725	0	167	0	25	0	3
21000. 22000.	7	28	65	174	0	725	0	167	0	25	0	3
22000. 23000.	9	21	0	109	723	725	0	167	0	25	0	3
23000. 24000.	3	12	96	109	0	2	141	167	23	25	0	3
24000. 25000.	4	9	13	13	2	2	26	26	0	2	3	3
25000. 26000.	2	5	0	0	0	0	0	0	2	2	0	0
26000. 27000.	0	3	0	0	0	0	0	0	0	0	0	0
27000. 28000.	1	3	0	0	0	0	0	0	0	0	0	0
28000. 29000.	0	2	0	0	0	0	0	0	0	0	0	0
29000. 30000.	2	2	0	0	0	0	0	0	0	0	0	0
30000. 31000.	0	0	0	0	0	0	0	0	0	0	0	0
31000. 32000.	0	0	0	0	0	0	0	0	0	0	0	0
32000. 33000.	0	0	0	0	0	0	0	0	0	0	0	0
33000. 34000.	0	0	0	0	0	0	0	0	0	0	0	0
34000. 35000.	0	0	0	0	0	0	0	0	0	0	0	0
35000. 36000.	0	0	0	0	0	0	0	0	0	0	0	0
36000. 37000.	0	0	0	0	0	0	0	0	0	0	0	0
37000. 38000.	0	0	0	0	0	0	0	0	0	0	0	0
38000. 39000.	0	0	0	0	0	0	0	0	0	0	0	0
39000. 40000.	0	0	0	0	0	0	0	0	0	0	0	0
40000. 41000.	0	0	0	0	0	0	0	0	0	0	0	0
41000. 42000.	0	0	0	0	0	0	0	0	0	0	0	0
42000. 43000.	0	0	0	0	0	0	0	0	0	0	0	0
43000. 44000.	0	0	0	0	0	0	0	0	0	0	0	0
ABOVE 44000.	0	0	0	0	0	0	0	0	0	0	0	0

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 18

JOB TITLE BS1

SPECTRUM SUMMATION FOR A TOTAL OF 2528 FLIGHTS AND 45807 CYCLES

VALLEY/PEAK RATIO RANGE			-.90		-.80		-.70		-.60		-.50		-.40
BELOW OR EQUAL	2000.	0	51	0	98	0	791	0	231	0	303	0	47
2000.	3000.	0	51	0	98	0	791	0	231	0	303	0	47
3000.	4000.	0	51	0	98	0	791	0	231	0	303	0	47
4000.	5000.	0	51	0	98	0	791	0	231	0	303	0	47
5000.	6000.	0	51	0	98	0	791	0	231	0	303	0	47
6000.	7000.	0	51	0	98	0	791	0	231	0	303	0	47
7000.	8000.	0	51	0	98	0	791	0	231	0	303	33	47
8000.	9000.	0	51	0	98	0	791	26	231	0	303	0	14
9000.	10000.	44	51	0	98	0	791	0	205	0	303	0	14
10000.	11000.	0	7	0	98	0	791	0	205	0	303	0	14
11000.	12000.	0	7	0	98	0	791	0	205	0	303	0	14
12000.	13000.	0	7	0	98	0	791	0	205	0	303	0	14
13000.	14000.	0	7	0	98	0	791	0	205	0	303	0	14
14000.	15000.	0	7	0	98	0	791	0	205	0	303	0	14
15000.	16000.	0	7	0	98	0	791	0	205	0	303	0	14
16000.	17000.	0	7	0	98	0	791	0	205	0	303	0	14
17000.	18000.	0	7	0	98	0	791	0	205	0	303	0	14
18000.	19000.	0	7	0	98	0	791	14	205	0	303	0	14
19000.	20000.	0	7	0	98	0	791	0	191	199	303	0	14
20000.	21000.	0	7	53	98	0	791	0	191	43	104	0	14
21000.	22000.	5	7	0	45	164	791	0	191	57	61	0	14
22000.	23000.	0	2	12	45	616	627	0	191	0	4	12	14
23000.	24000.	0	2	33	33	4	11	167	191	0	4	2	2
24000.	25000.	0	2	0	0	7	7	22	24	0	4	0	0
25000.	26000.	2	2	0	0	0	0	2	2	3	4	0	0
26000.	27000.	0	0	0	0	0	0	0	0	0	1	0	0
27000.	28000.	0	0	0	0	0	0	0	0	1	1	0	0
28000.	29000.	0	0	0	0	0	0	0	0	0	0	0	0
29000.	30000.	0	0	0	0	0	0	0	0	0	0	0	0
30000.	31000.	0	0	0	0	0	0	0	0	0	0	0	0
31000.	32000.	0	0	0	0	0	0	0	0	0	0	0	0
32000.	33000.	0	0	0	0	0	0	0	0	0	0	0	0
33000.	34000.	0	0	0	0	0	0	0	0	0	0	0	0
34000.	35000.	0	0	0	0	0	0	0	0	0	0	0	0
35000.	36000.	0	0	0	0	0	0	0	0	0	0	0	0
36000.	37000.	0	0	0	0	0	0	0	0	0	0	0	0
37000.	38000.	0	0	0	0	0	0	0	0	0	0	0	0
38000.	39000.	0	0	0	0	0	0	0	0	0	0	0	0
39000.	40000.	0	0	0	0	0	0	0	0	0	0	0	0
40000.	41000.	0	0	0	0	0	0	0	0	0	0	0	0
41000.	42000.	0	0	0	0	0	0	0	0	0	0	0	0
42000.	43000.	0	0	0	0	0	0	0	0	0	0	0	0
43000.	44000.	0	0	0	0	0	0	0	0	0	0	0	0
ABOVE	44000.	0	0	0	0	0	0	0	0	0	0	0	0

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM   PAGE 19

JOB TITLE BS1

SPECTRUM SUMMATION FOR A TOTAL OF 2528 FLIGHTS AND 45807 CYCLES

VALLEY/PEAK RATIO RANGE		-.30		-.20		-.10		.00		.10		.20
BELOW OR EQUAL 2000.	0	52	0	0	0	47	0	4	0	89	0	30
2000. 3000.	0	52	0	0	0	47	0	4	0	89	0	30
3000. 4000.	0	52	0	0	0	47	0	4	0	89	0	30
4000. 5000.	0	52	0	0	0	47	0	4	78	89	0	30
5000. 6000.	0	52	0	0	44	47	0	4	0	11	0	30
6000. 7000.	52	52	0	0	0	3	0	4	0	11	2	30
7000. 8000.	0	0	0	0	0	3	0	4	2	11	9	28
8000. 9000.	0	0	0	0	0	3	1	4	3	9	4	19
9000. 10000.	0	0	0	0	1	3	1	3	1	6	3	15
10000. 11000.	0	0	0	0	1	2	0	2	0	5	2	12
11000. 12000.	0	0	0	0	0	1	0	2	3	5	7	10
12000. 13000.	0	0	0	0	0	1	1	2	2	2	2	3
13000. 14000.	0	0	0	0	1	1	1	1	0	0	1	1
14000. 15000.	0	0	0	0	0	0	0	0	0	0	0	0
15000. 16000.	0	0	0	0	0	0	0	0	0	0	0	0
16000. 17000.	0	0	0	0	0	0	0	0	0	0	0	0
17000. 18000.	0	0	0	0	0	0	0	0	0	0	0	0
18000. 19000.	0	0	0	0	0	0	0	0	0	0	0	0
19000. 20000.	0	0	0	0	0	0	0	0	0	0	0	0
20000. 21000.	0	0	0	0	0	0	0	0	0	0	0	0
21000. 22000.	0	0	0	0	0	0	0	0	0	0	0	0
22000. 23000.	0	0	0	0	0	0	0	0	0	0	0	0
23000. 24000.	0	0	0	0	0	0	0	0	0	0	0	0
24000. 25000.	0	0	0	0	0	0	0	0	0	0	0	0
25000. 26000.	0	0	0	0	0	0	0	0	0	0	0	0
26000. 27000.	0	0	0	0	0	0	0	0	0	0	0	0
27000. 28000.	0	0	0	0	0	0	0	0	0	0	0	0
28000. 29000.	0	0	0	0	0	0	0	0	0	0	0	0
29000. 30000.	0	0	0	0	0	0	0	0	0	0	0	0
30000. 31000.	0	0	0	0	0	0	0	0	0	0	0	0
31000. 32000.	0	0	0	0	0	0	0	0	0	0	0	0
32000. 33000.	0	0	0	0	0	0	0	0	0	0	0	0
33000. 34000.	0	0	0	0	0	0	0	0	0	0	0	0
34000. 35000.	0	0	0	0	0	0	0	0	0	0	0	0
35000. 36000.	0	0	0	0	0	0	0	0	0	0	0	0
36000. 37000.	0	0	0	0	0	0	0	0	0	0	0	0
37000. 38000.	0	0	0	0	0	0	0	0	0	0	0	0
38000. 39000.	0	0	0	0	0	0	0	0	0	0	0	0
39000. 40000.	0	0	0	0	0	0	0	0	0	0	0	0
40000. 41000.	0	0	0	0	0	0	0	0	0	0	0	0
41000. 42000.	0	0	0	0	0	0	0	0	0	0	0	0
42000. 43000.	0	0	0	0	0	0	0	0	0	0	0	0
43000. 44000.	0	0	0	0	0	0	0	0	0	0	0	0
ABOVE 44000.	0	0	0	0	0	0	0	0	0	0	0	0

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 20

JOB TITLE BS1

SPECTRUM SUMMATION FOR A TOTAL OF 2528 FLIGHTS AND 45807 CYCLES

VALLEY/PEAK RATIO RANGE				.30		.40		.50		.60		.70	.80
BELOW OR EQUAL	2000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210
2000.	3000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210
3000.	4000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210
4000.	5000.	145	235	0	500	543	3897	8015	13076	10596	12294	1210	1210
5000.	6000.	6	90	137	500	1655	3354	982	5061	1522	1698	0	0
6000.	7000.	26	84	91	363	969	1699	3664	4079	175	176	0	0
7000.	8000.	14	58	136	272	572	730	388	415	1	1	0	0
8000.	9000.	31	44	108	136	119	158	25	27	0	0	0	0
9000.	10000.	12	13	20	28	25	39	2	2	0	0	0	0
10000.	11000.	0	1	3	8	3	14	0	0	0	0	0	0
11000.	12000.	1	1	2	5	9	11	0	0	0	0	0	0
12000.	13000.	0	0	1	3	2	2	0	0	0	0	0	0
13000.	14000.	0	0	2	2	0	0	0	0	0	0	0	0
14000.	15000.	0	0	0	0	0	0	0	0	0	0	0	0
15000.	16000.	0	0	0	0	0	0	0	0	0	0	0	0
16000.	17000.	0	0	0	0	0	0	0	0	0	0	0	0
17000.	18000.	0	0	0	0	0	0	0	0	0	0	0	0
18000.	19000.	0	0	0	0	0	0	0	0	0	0	0	0
19000.	20000.	0	0	0	0	0	0	0	0	0	0	0	0
20000.	21000.	0	0	0	0	0	0	0	0	0	0	0	0
21000.	22000.	0	0	0	0	0	0	0	0	0	0	0	0
22000.	23000.	0	0	0	0	0	0	0	0	0	0	0	0
23000.	24000.	0	0	0	0	0	0	0	0	0	0	0	0
24000.	25000.	0	0	0	0	0	0	0	0	0	0	0	0
25000.	26000.	0	0	0	0	0	0	0	0	0	0	0	0
26000.	27000.	0	0	0	0	0	0	0	0	0	0	0	0
27000.	28000.	0	0	0	0	0	0	0	0	0	0	0	0
28000.	29000.	0	0	0	0	0	0	0	0	0	0	0	0
29000.	30000.	0	0	0	0	0	0	0	0	0	0	0	0
30000.	31000.	0	0	0	0	0	0	0	0	0	0	0	0
31000.	32000.	0	0	0	0	0	0	0	0	0	0	0	0
32000.	33000.	0	0	0	0	0	0	0	0	0	0	0	0
33000.	34000.	0	0	0	0	0	0	0	0	0	0	0	0
34000.	35000.	0	0	0	0	0	0	0	0	0	0	0	0
35000.	36000.	0	0	0	0	0	0	0	0	0	0	0	0
36000.	37000.	0	0	0	0	0	0	0	0	0	0	0	0
37000.	38000.	0	0	0	0	0	0	0	0	0	0	0	0
38000.	39000.	0	0	0	0	0	0	0	0	0	0	0	0
39000.	40000.	0	0	0	0	0	0	0	0	0	0	0	0
40000.	41000.	0	0	0	0	0	0	0	0	0	0	0	0
41000.	42000.	0	0	0	0	0	0	0	0	0	0	0	0
42000.	43000.	0	0	0	0	0	0	0	0	0	0	0	0
43000.	44000.	0	0	0	0	0	0	0	0	0	0	0	0
ABOVE	44000.	0	0	0	0	0	0	0	0	0	0	0	0

JOB TITLE BS1

SPECTRUM SUMMATION FOR A TOTAL OF 2528 FLIGHTS AND 45807 CYCLES

VALLEY/PEAK RATIO			.90		1.00		TOTAL	
RANGE								
BELOW OR EQUAL 2000.	0	0	0	0	0	11665	0	45807
2000. 3000.	0	0	0	0	0	11665	0	45807
3000. 4000.	0	0	0	0	0	11665	0	45807
4000. 5000.	0	0	0	0	1504	11665	22091	45807
5000. 6000.	0	0	0	0	143	10161	4489	23716
6000. 7000.	0	0	0	0	5262	10018	10241	19227
7000. 8000.	0	0	0	0	3459	4756	4614	8986
8000. 9000.	0	0	0	0	1045	1297	1362	4372
9000. 10000.	0	0	0	0	207	252	316	3010
10000. 11000.	0	0	0	0	36	45	45	2694
11000. 12000.	0	0	0	0	8	9	52	2649
12000. 13000.	0	0	0	0	1	1	9	2597
13000. 14000.	0	0	0	0	0	0	19	2588
14000. 15000.	0	0	0	0	0	0	0	2569
15000. 16000.	0	0	0	0	0	0	12	2569
16000. 17000.	0	0	0	0	0	0	7	2557
17000. 18000.	0	0	0	0	0	0	0	2550
18000. 19000.	0	0	0	0	0	0	19	2550
19000. 20000.	0	0	0	0	0	0	199	2531
20000. 21000.	0	0	0	0	0	0	101	2332
21000. 22000.	0	0	0	0	0	0	298	2231
22000. 23000.	0	0	0	0	0	0	1372	1933
23000. 24000.	0	0	0	0	0	0	469	561
24000. 25000.	0	0	0	0	0	0	77	92
25000. 26000.	0	0	0	0	0	0	11	15
26000. 27000.	0	0	0	0	0	0	0	4
27000. 28000.	0	0	0	0	0	0	2	4
28000. 29000.	0	0	0	0	0	0	0	2
29000. 30000.	0	0	0	0	0	0	2	2
30000. 31000.	0	0	0	0	0	0	0	0
31000. 32000.	0	0	0	0	0	0	0	0
32000. 33000.	0	0	0	0	0	0	0	0
33000. 34000.	0	0	0	0	0	0	0	0
34000. 35000.	0	0	0	0	0	0	0	0
35000. 36000.	0	0	0	0	0	0	0	0
36000. 37000.	0	0	0	0	0	0	0	0
37000. 38000.	0	0	0	0	0	0	0	0
38000. 39000.	0	0	0	0	0	0	0	0
39000. 40000.	0	0	0	0	0	0	0	0
40000. 41000.	0	0	0	0	0	0	0	0
41000. 42000.	0	0	0	0	0	0	0	0
42000. 43000.	0	0	0	0	0	0	0	0
43000. 44000.	0	0	0	0	0	0	0	0
ABOVE 44000.	0	0	0	0	0	0	0	0

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 22

JOB TITLE **BS1**

SPECTRUM SUMMATION FOR A TOTAL OF 2528 FLIGHTS AND 45807 CYCLES

VALLEY/PEAK RATIO	PEAK		-1.50		-1.40		-1.30		-1.20		-1.10		-1.00
BELOW OR EQUAL	-8000.	0	71	0	174	0	725	0	189	0	25	0	3
-8000.	-7000.	0	71	0	174	0	725	0	189	0	25	0	3
-7000.	-6000.	0	71	0	174	0	725	0	189	0	25	0	3
-6000.	-5000.	0	71	0	174	0	725	0	189	0	25	0	3
-5000.	-4000.	0	71	0	174	0	725	0	189	0	25	0	3
-4000.	-3000.	0	71	0	174	0	725	0	189	0	25	0	3
-3000.	-2000.	0	71	0	174	0	725	0	189	0	25	0	3
-2000.	-1000.	0	71	0	174	0	725	0	189	0	25	0	3
-1000.	0.	0	71	0	174	0	725	0	189	0	25	0	3
0.	1000.	0	71	0	174	0	725	0	189	0	25	0	3
1000.	2000.	0	71	0	174	0	725	0	189	0	25	0	3
2000.	3000.	0	71	0	174	0	725	0	189	0	25	0	3
3000.	4000.	0	71	0	174	0	725	0	189	0	25	0	3
4000.	5000.	0	71	0	174	0	725	0	189	0	25	0	3
5000.	6000.	55	71	0	174	0	725	0	189	0	25	0	3
6000.	7000.	0	16	0	174	0	725	22	189	0	25	0	3
7000.	8000.	0	16	0	174	0	725	0	167	0	25	0	3
8000.	9000.	15	16	64	174	0	725	0	167	0	25	0	3
9000.	10000.	1	1	110	110	0	725	0	167	0	25	0	3
10000.	11000.	0	0	0	0	723	725	0	167	0	25	0	3
11000.	12000.	0	0	0	0	2	2	162	167	0	25	0	3
12000.	13000.	0	0	0	0	0	0	5	5	23	25	0	3
13000.	14000.	0	0	0	0	0	0	0	0	2	2	3	3
14000.	15000.	0	0	0	0	0	0	0	0	0	0	0	0
15000.	16000.	0	0	0	0	0	0	0	0	0	0	0	0
16000.	17000.	0	0	0	0	0	0	0	0	0	0	0	0
17000.	18000.	0	0	0	0	0	0	0	0	0	0	0	0
18000.	19000.	0	0	0	0	0	0	0	0	0	0	0	0
19000.	20000.	0	0	0	0	0	0	0	0	0	0	0	0
20000.	21000.	0	0	0	0	0	0	0	0	0	0	0	0
21000.	22000.	0	0	0	0	0	0	0	0	0	0	0	0
22000.	23000.	0	0	0	0	0	0	0	0	0	0	0	0
23000.	24000.	0	0	0	0	0	0	0	0	0	0	0	0
24000.	25000.	0	0	0	0	0	0	0	0	0	0	0	0
25000.	26000.	0	0	0	0	0	0	0	0	0	0	0	0
26000.	27000.	0	0	0	0	0	0	0	0	0	0	0	0
27000.	28000.	0	0	0	0	0	0	0	0	0	0	0	0
28000.	29000.	0	0	0	0	0	0	0	0	0	0	0	0
29000.	30000.	0	0	0	0	0	0	0	0	0	0	0	0
30000.	31000.	0	0	0	0	0	0	0	0	0	0	0	0
31000.	32000.	0	0	0	0	0	0	0	0	0	0	0	0
32000.	33000.	0	0	0	0	0	0	0	0	0	0	0	0
33000.	34000.	0	0	0	0	0	0	0	0	0	0	0	0
34000.	35000.	0	0	0	0	0	0	0	0	0	0	0	0
35000.	36000.	0	0	0	0	0	0	0	0	0	0	0	0

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 23

JOB TITLE BS1

SPECTRUM SUMMATION FOR A TOTAL OF 2528 FLIGHTS AND 45807 CYCLES

VALLEY/PEAK RATIO	-1.50		-1.40		-1.30		-1.20		-1.10		-1.00	
PEAK												
36000. 37000.	0	0	0	0	0	0	0	0	0	0	0	0
37000. 38000.	0	0	0	0	0	0	0	0	0	0	0	0
38000. 39000.	0	0	0	0	0	0	0	0	0	0	0	0
ABOVE 39000.	0	0	0	0	0	0	0	0	0	0	0	0

VALLEY/PEAK RATIO	-.90		-.80		-.70		-.60		-.50		-.40	
PEAK												
BELOW OR EQUAL -8000.	0	51	0	98	0	791	0	231	0	303	0	47
-8000. -7000.	0	51	0	98	0	791	0	231	0	303	0	47
-7000. -6000.	0	51	0	98	0	791	0	231	0	303	0	47
-6000. -5000.	0	51	0	98	0	791	0	231	0	303	0	47
-5000. -4000.	0	51	0	98	0	791	0	231	0	303	0	47
-4000. -3000.	0	51	0	98	0	791	0	231	0	303	0	47
-3000. -2000.	0	51	0	98	0	791	0	231	0	303	0	47
-2000. -1000.	0	51	0	98	0	791	0	231	0	303	0	47
-1000. 0.	0	51	0	98	0	791	0	231	0	303	0	47
0. 1000.	0	51	0	98	0	791	0	231	0	303	0	47
1000. 2000.	0	51	0	98	0	791	0	231	0	303	0	47
2000. 3000.	0	51	0	98	0	791	0	231	0	303	0	47
3000. 4000.	0	51	0	98	0	791	0	231	0	303	0	47
4000. 5000.	0	51	0	98	0	791	0	231	0	303	0	47
5000. 6000.	44	51	0	98	0	791	26	231	0	303	33	47
6000. 7000.	0	7	0	98	0	791	0	205	0	303	0	14
7000. 8000.	0	7	0	98	0	791	0	205	0	303	0	14
8000. 9000.	0	7	0	98	0	791	0	205	0	303	0	14
9000. 10000.	0	7	0	98	0	791	0	205	0	303	0	14
10000. 11000.	5	7	53	98	0	791	14	205	0	303	0	14
11000. 12000.	0	2	0	45	0	791	0	191	0	303	0	14
12000. 13000.	2	2	45	45	665	791	0	191	199	303	0	14
13000. 14000.	0	0	0	0	119	126	0	191	43	104	0	14
14000. 15000.	0	0	0	0	7	7	167	191	57	61	0	14
15000. 16000.	0	0	0	0	0	0	24	24	0	4	12	14
16000. 17000.	0	0	0	0	0	0	0	0	3	4	2	2
17000. 18000.	0	0	0	0	0	0	0	0	1	1	0	0
18000. 19000.	0	0	0	0	0	0	0	0	0	0	0	0
19000. 20000.	0	0	0	0	0	0	0	0	0	0	0	0
20000. 21000.	0	0	0	0	0	0	0	0	0	0	0	0
21000. 22000.	0	0	0	0	0	0	0	0	0	0	0	0
22000. 23000.	0	0	0	0	0	0	0	0	0	0	0	0
23000. 24000.	0	0	0	0	0	0	0	0	0	0	0	0
24000. 25000.	0	0	0	0	0	0	0	0	0	0	0	0
25000. 26000.	0	0	0	0	0	0	0	0	0	0	0	0
26000. 27000.	0	0	0	0	0	0	0	0	0	0	0	0
27000. 28000.	0	0	0	0	0	0	0	0	0	0	0	0
28000. 29000.	0	0	0	0	0	0	0	0	0	0	0	0
29000. 30000.	0	0	0	0	0	0	0	0	0	0	0	0
30000. 31000.	0	0	0	0	0	0	0	0	0	0	0	0
31000. 32000.	0	0	0	0	0	0	0	0	0	0	0	0
32000. 33000.	0	0	0	0	0	0	0	0	0	0	0	0

JOB TITLE B$1

SPECTRUM SUMMATION FOR A TOTAL OF 2528 FLIGHTS AND 45807 CYCLES

VALLEY/PEAK RATIO / PEAK		-.90		-.80		-.70		-.60		-.50		-.40
33000. 34000.	0	0	0	0	0	0	0	0	0	0	0	0
34000. 35000.	0	0	0	0	0	0	0	0	0	0	0	0
35000. 36000.	0	0	0	0	0	0	0	0	0	0	0	0
36000. 37000.	0	0	0	0	0	0	0	0	0	0	0	0
37000. 38000.	0	0	0	0	0	0	0	0	0	0	0	0
38000. 39000.	0	0	0	0	0	0	0	0	0	0	0	0
ABOVE 39000.	0	0	0	0	0	0	0	0	0	0	0	0

VALLEY/PEAK RATIO / PEAK		-.30		-.20		-.10		.00		.10		.20
BELOW OR EQUAL -8000.	0	52	0	0	0	47	0	4	0	89	0	30
-8000. -7000.	0	52	0	0	0	47	0	4	0	89	0	30
-7000. -6000.	0	52	0	0	0	47	0	4	0	89	0	30
-6000. -5000.	0	52	0	0	0	47	0	4	0	89	0	30
-5000. -4000.	0	52	0	0	0	47	0	4	0	89	0	30
-4000. -3000.	0	52	0	0	0	47	0	4	0	89	0	30
-3000. -2000.	0	52	0	0	0	47	0	4	0	89	0	30
-2000. -1000.	0	52	0	0	0	47	0	4	0	89	0	30
-1000. 0.	0	52	0	0	0	47	0	4	0	89	0	30
0. 1000.	0	52	0	0	0	47	0	4	0	89	0	30
1000. 2000.	0	52	0	0	0	47	0	4	0	89	0	30
2000. 3000.	0	52	0	0	0	47	0	4	0	89	0	30
3000. 4000.	0	52	0	0	0	47	0	4	0	89	0	30
4000. 5000.	0	52	0	0	0	47	0	4	0	89	0	30
5000. 6000.	52	52	0	0	44	47	0	4	78	89	0	30
6000. 7000.	0	0	0	0	0	3	0	4	0	11	0	30
7000. 8000.	0	0	0	0	1	3	0	4	2	11	2	30
8000. 9000.	0	0	0	0	0	2	1	4	1	9	7	28
9000. 10000.	0	0	0	0	1	2	1	3	3	8	5	21
10000. 11000.	0	0	0	0	0	1	0	2	0	5	1	16
11000. 12000.	0	0	0	0	0	1	0	2	0	5	0	15
12000. 13000.	0	0	0	0	1	1	2	2	4	5	7	15
13000. 14000.	0	0	0	0	0	0	0	0	1	1	5	8
14000. 15000.	0	0	0	0	0	0	0	0	0	0	2	3
15000. 16000.	0	0	0	0	0	0	0	0	0	0	1	1
16000. 17000.	0	0	0	0	0	0	0	0	0	0	0	0
17000. 18000.	0	0	0	0	0	0	0	0	0	0	0	0
18000. 19000.	0	0	0	0	0	0	0	0	0	0	0	0
19000. 20000.	0	0	0	0	0	0	0	0	0	0	0	0
20000. 21000.	0	0	0	0	0	0	0	0	0	0	0	0
21000. 22000.	0	0	0	0	0	0	0	0	0	0	0	0
22000. 23000.	0	0	0	0	0	0	0	0	0	0	0	0
23000. 24000.	0	0	0	0	0	0	0	0	0	0	0	0
24000. 25000.	0	0	0	0	0	0	0	0	0	0	0	0
25000. 26000.	0	0	0	0	0	0	0	0	0	0	0	0
26000. 27000.	0	0	0	0	0	0	0	0	0	0	0	0
27000. 28000.	0	0	0	0	0	0	0	0	0	0	0	0
28000. 29000.	0	0	0	0	0	0	0	0	0	0	0	0
29000. 30000.	0	0	0	0	0	0	0	0	0	0	0	0

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 25

JOB TITLE BS1

SPECTRUM SUMMATION FOR A TOTAL OF 2528 FLIGHTS AND 45807 CYCLES

VALLEY/PEAK RATIO PEAK			-.30		-.20		-.10		.00		.10		.20
30000. 31000.	0	0	0	0	0	0	0	0	0	0	0	0	
31000. 32000.	0	0	0	0	0	0	0	0	0	0	0	0	
32000. 33000.	0	0	0	0	0	0	0	0	0	0	0	0	
33000. 34000.	0	0	0	0	0	0	0	0	0	0	0	0	
34000. 35000.	0	0	0	0	0	0	0	0	0	0	0	0	
35000. 36000.	0	0	0	0	0	0	0	0	0	0	0	0	
36000. 37000.	0	0	0	0	0	0	0	0	0	0	0	0	
37000. 38000.	0	0	0	0	0	0	0	0	0	0	0	0	
38000. 39000.	0	0	0	0	0	0	0	0	0	0	0	0	
ABOVE 39000.	0	0	0	0	0	0	0	0	0	0	0	0	

VALLEY/PEAK RATIO PEAK			.30		.40		.50		.60		.70		.80
BELOW OR EQUAL -8000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
-8000. -7000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
-7000. -6000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
-6000. -5000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
-5000. -4000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
-4000. -3000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
-3000. -2000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
-2000. -1000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
-1000. 0.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
0. 1000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
1000. 2000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
2000. 3000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
3000. 4000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
4000. 5000.	0	235	0	500	0	3897	0	13076	0	12294	0	1210	
5000. 6000.	145	235	0	500	0	3897	0	13076	0	12294	0	1210	
6000. 7000.	0	90	0	500	0	3897	0	13076	0	12294	0	1210	
7000. 8000.	5	90	32	500	170	3897	0	13076	0	12294	0	1210	
8000. 9000.	19	85	68	468	398	3727	1339	13076	0	12294	0	1210	
9000. 10000.	22	66	103	400	906	3329	2920	11737	0	12294	0	1210	
10000. 11000.	2	44	40	297	709	2423	1685	8817	3	12294	0	1210	
11000. 12000.	8	42	27	257	351	1714	569	7132	128	12291	0	1210	
12000. 13000.	25	34	110	230	739	1363	2773	6563	1846	12163	0	1210	
13000. 14000.	8	9	93	120	369	624	1554	3790	8911	10317	0	1210	
14000. 15000.	0	1	12	27	121	255	1622	2236	372	1406	1136	1210	
15000. 16000.	1	1	10	15	55	134	529	614	885	1034	11	74	
16000. 17000.	0	0	1	5	1	79	23	85	121	149	63	63	
17000. 18000.	0	0	1	4	53	78	44	62	24	28	0	0	
18000. 19000.	0	0	1	3	12	25	15	18	0	4	0	0	
19000. 20000.	0	0	0	2	8	13	3	3	3	4	0	0	
20000. 21000.	0	0	0	2	5	5	0	0	1	1	0	0	
21000. 22000.	0	0	1	2	0	0	0	0	0	0	0	0	
22000. 23000.	0	0	1	1	0	0	0	0	0	0	0	0	
23000. 24000.	0	0	0	0	0	0	0	0	0	0	0	0	
24000. 25000.	0	0	0	0	0	0	0	0	0	0	0	0	
25000. 26000.	0	0	0	0	0	0	0	0	0	0	0	0	
26000. 27000.	0	0	0	0	0	0	0	0	0	0	0	0	

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 26

JOB TITLE BS1

SPECTRUM SUMMATION FOR A TOTAL OF 2528 FLIGHTS AND 45807 CYCLES

VALLEY/PEAK RATIO PEAK			.30		.40		.50		.60		.70		.80
27000. 28000.	0	0	0	0	0	0	0	0	0	0	0	0	
28000. 29000.	0	0	0	0	0	0	0	0	0	0	0	0	
29000. 30000.	0	0	0	0	0	0	0	0	0	0	0	0	
30000. 31000.	0	0	0	0	0	0	0	0	0	0	0	0	
31000. 32000.	0	0	0	0	0	0	0	0	0	0	0	0	
32000. 33000.	0	0	0	0	0	0	0	0	0	0	0	0	
33000. 34000.	0	0	0	0	0	0	0	0	0	0	0	0	
34000. 35000.	0	0	0	0	0	0	0	0	0	0	0	0	
35000. 36000.	0	0	0	0	0	0	0	0	0	0	0	0	
36000. 37000.	0	0	0	0	0	0	0	0	0	0	0	0	
37000. 38000.	0	0	0	0	0	0	0	0	0	0	0	0	
38000. 39000.	0	0	0	0	0	0	0	0	0	0	0	0	
ABOVE 39000.	0	0	0	0	0	0	0	0	0	0	0	0	

VALLEY/PEAK RATIO PEAK			.90		1.00		TOTAL	
BELOW OR EQUAL -8000.	0	0	0	0	0	11665	0	45807
-8000. -7000.	0	0	0	0	0	11665	0	45807
-7000. -6000.	0	0	0	0	0	11665	0	45807
-6000. -5000.	0	0	0	0	0	11665	0	45807
-5000. -4000.	0	0	0	0	5151	11665	5151	45807
-4000. -3000.	0	0	0	0	6386	6514	6386	40656
-3000. -2000.	0	0	0	0	110	128	110	34270
-2000. -1000.	0	0	0	0	18	18	18	34160
-1000. 0.	0	0	0	0	0	0	0	34142
0. 1000.	0	0	0	0	0	0	0	34142
1000. 2000.	0	0	0	0	0	0	0	34142
2000. 3000.	0	0	0	0	0	0	0	34142
3000. 4000.	0	0	0	0	0	0	0	34142
4000. 5000.	0	0	0	0	0	0	0	34142
5000. 6000.	0	0	0	0	0	0	499	34142
6000. 7000.	0	0	0	0	0	0	0	33643
7000. 8000.	0	0	0	0	0	0	212	33643
8000. 9000.	0	0	0	0	0	0	1912	33431
9000. 10000.	0	0	0	0	0	0	4795	31519
10000. 11000.	0	0	0	0	0	0	2676	26724
11000. 12000.	0	0	0	0	0	0	1111	24048
12000. 13000.	0	0	0	0	0	0	6423	22937
13000. 14000.	0	0	0	0	0	0	11103	16514
14000. 15000.	0	0	0	0	0	0	3496	5411
15000. 16000.	0	0	0	0	0	0	1528	1915
16000. 17000.	0	0	0	0	0	0	214	387
17000. 18000.	0	0	0	0	0	0	123	173
18000. 19000.	0	0	0	0	0	0	28	50
19000. 20000.	0	0	0	0	0	0	14	22
20000. 21000.	0	0	0	0	0	0	6	8
21000. 22000.	0	0	0	0	0	0	1	2
22000. 23000.	0	0	0	0	0	0	1	1
23000. 24000.	0	0	0	0	0	0	0	0

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 27

JOB TITLE BS1

SPECTRUM SUMMATION FOR A TOTAL OF 2528 FLIGHTS AND 45807 CYCLES

VALLEY/PEAK RATIO			.90		1.00			TOTAL	
PEAK									
24000.	25000.	0	0	0	0	0	0	0	0
25000.	26000.	0	0	0	0	0	0	0	0
26000.	27000.	0	0	0	0	0	0	0	0
27000.	28000.	0	0	0	0	0	0	0	0
28000.	29000.	0	0	0	0	0	0	0	0
29000.	30000.	0	0	0	0	0	0	0	0
30000.	31000.	0	0	0	0	0	0	0	0
31000.	32000.	0	0	0	0	0	0	0	0
32000.	33000.	0	0	0	0	0	0	0	0
33000.	34000.	0	0	0	0	0	0	0	0
34000.	35000.	0	0	0	0	0	0	0	0
35000.	36000.	0	0	0	0	0	0	0	0
36000.	37000.	0	0	0	0	0	0	0	0
37000.	38000.	0	0	0	0	0	0	0	0
38000.	39000.	0	0	0	0	0	0	0	0
ABOVE	39000.	0	0	0	0	0	0	0	0

SPECTRUM LOADING SEQUENCE GENERATION PROGRAM PAGE 28

JOB TITLE BS1

SPECTRUM SUMMATION FOR A TOTAL OF 2528 FLIGHTS AND 45807 CYCLES

FLIGHT TYPE	NUMBER FLIGHTS	NUMBER CYCLES	FLIGHT TYPE	NUMBER FLIGHTS	NUMBER CYCLES
1	1112	21722	2	1416	24085

THE 20 HIGHEST PEAKS

PEAK	FLIGHT	PEAK	FLIGHT	PEAK	FLIGHT	PEAK	FLIGHT	PEAK	FLIGHT	PEAK	FLIGHT
22113.80	1833	21969.65	1988	20885.40	443	20885.40	991	20746.95	1152	20746.95	1449
20382.95	181	20051.95	356	19657.00	53	19657.00	1020	19657.00	1548	19657.00	1911
19543.75	465	19524.25	1061	19524.25	1248	19524.25	1999	19524.25	2489	19524.25	2512
19158.65	788	19158.65	1217								

REFERENCES

1. Abelkis, P. R. and Bobovski, W. P.: "Fatigue Strength Design and Analysis of Aircraft Structures. Part II - Fatigue Life Analysis Computer Program - User's Manual," AFFDL-TR-66-197, Part II, March 1967.

2. Abelkis, P. R.: "Effect of Transport/Bomber Loads Spectrum on Crack Growth," AFFDL-TR-78-134, November 1978.

3. Military Specification, "Airplane Strength and Rigidity Reliability Requirements, Repeated Loads and Fatigue," MIL-A-008866, 22 August 1975.

4. Control Data Corp. publication 60342500, "Control Data Cyber 70 6000/7600 Series Computer Systems UPDATE Reference Manual," Revision C, 1973.

APPENDIX A

PROGRAM A6PA CHANGES

This appendix outlines the changes made to program A6PA since its publication as program 16PA in Reference 1. The listing given in Section V reflects these changes. The changes consist of a minor internal program adjustment in segment spectra summation procedure and an addition to calculate gust airplane side load factor and vertical tail load spectra.

1. Spectra Summation

The need to calculate GAG cycle damge ($I4 \neq 0$) in order to perform segment spectra summation ($IW4 = 1$) was eliminated. Segment spectra summation can be obtained even when GAG cycle damage is not calculated ($I4 = 0$).

2. Airplane Side Load Factor and Vertical Tail Load Spectra Calculation

The calculation of airplane side load factor and vertical tail load spectra due to lateral gusts were added to the program. The calculations are performed and obtained in the following manner:

Cumulative frequency spectrum is calculated as,

$$\Sigma n(\Delta y)_j = \left(N_{0_1} P_1 e^{-\Delta y_j / b_1 \bar{A}} + N_{0_2} P_2 e^{-\Delta y_j / b_2 \bar{A}} \right)_T$$

Airplane Side Load Factor, M3 = 14

Δy = airplane side load factor

$\bar{A} = \left[V_e S_w K g_T \big/ 498W \right] \left[C_{y_{\beta_T}} + C_{y_{\beta_{TO}}} \right]$

= calculated by the program.

$K_{g_T} = .88 \mu_{g_T} \Big/ \left(5.3 + \mu_{g_T} \right)$

$\mu_{g_T} = \left(2W / (\rho/\rho_0) \rho_0 \bar{C}_t \, a_t \, S_t g \right) (K / l_t)^2$

$K = \sqrt{I_{yaw} / W}$ = radius of gyration in yaw, in.

V_e = airplane speed, KEAS

S_w = wing area, ft^2

W = airplane weight, lbs.

$Cy^E_{\beta_T}$ = side force coefficient, tail contribution;

per radian = $Cy^R_{\beta_T} \left(C_{L_{\alpha_T}}\right)^{E/R} F_v$

F_v = fuselage bending factor due to tail lift

$Cy_{\beta_{TO}}$ = side force coefficient, tail off; per radian

ρ = air density at altitude, $slugs/ft^2$

ρ_0 = air density at sea level, $slugs/ft^2$

$\overline{C}_t$ = mean geometric chord of vertical surface, ft.

a_t = lift curve slope of vertical tail; per radian = $C_{L_{\alpha_V}}$

$$= Cy_{\beta_T} \left[\frac{S_w}{S_V\left(1+\frac{d\sigma}{d\beta}\right)}\right]$$

S_t = area of vertical tail, ft^2

I_{yaw} = yaw moment of inertia, $lb\text{-}in^2$

l_t = distance from airplane c.g. to lift center of vertical surface, in.

Vertical Tail Side Load, M3 = 15

$$\Sigma n(\Delta y_j) = \left(N_{0_1} P_1 e^{-\Delta y_j/b_1\overline{A}} + N_{0_2} P_2 e^{-\Delta y_j/b_2\overline{A}}\right)_T$$

where,

Δy = vertical tail side load, lbs.

$\overline{A}$ = $\left(V_e a_t S_t K_{g_t}/498\right)$

= calculated by the program

V_e, a_t, S_t, K_{g_t} = same as defined for M3 = 14

Input Data when M3 = 14 or 15

The input data in general is the same as in the vertical gust spectrum calculation M3 = 13. The main difference is in the data required to calculate $\bar{A}$, which is:

Parameter	M3 =		Load Sheet No.		
	14	15	New	Old(New Data)	Old
$\bar{C}_t$	✓	✓	III - 6.7		
S_t	✓	✓	III - 6.7		
I_{yaw}	✓	✓	III - 6.7		
$C_{y_{\beta T}}$	✓		III - 6.8		
$C_{y_{\beta TO}}$	✓		III - 6.9		
a_t	✓	✓		III - 6.2	
l_t	✓	✓		III - 6.6	
S_w	✓				I - 2
V_e	✓	✓			III - 6.3
W	✓	✓			III - 6.4
(ρ/ρ_0)	✓	✓			III - 6.5

The affected input data load sheets (new data or notes) are presented on pages 241 thru 246 .

When input data printout is requested, the new input data is printed under the following headers:

Parameter	Header
$\bar{C}_t$	VT Chord
S_t	VT Area
I_{yaw}	IYAW
$C_{y_{\beta_T}}$	SFC-T
$C_{y_{\beta_{TO}}}$	SFC-TO
a_t	SLOPE
l_t	SCALE OF TURBULENCE

The last two headers are the names of the other parameters (m and L) sharing the same location numbers.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET II-1
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______

69	70 71	73 77	80
R R	CASE	16 PA	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E	QUAN	LOC	±	VALUE	±	E
M3						M3					
Segm 1	1673					Segm 21	1693				
2	1674					22	1694				
3	1675					23	1695				
4	1676					24	1696				
5	1677					25	1697				
6	1678					26	1698				
7	1679					27	1699				
8	1680					28	1700				
9	1681					29	1701				
10	1682					30	1702				
11	1683					31	1703				
12	1684					32	1704				
13	1685					33	1705				
14	1686					34	1706				
15	1687					35	1707				
16	1688					36	1708				
17	1689					37	1709				
18	1690					38	1710				
19	1691					39	1711				
20	1692					40	1712				

M3 = FLAG WHICH SELECTS THE SPECTRUM CYCLE INPUT FORMAT; A VALUE (1 TO 15) MUST BE ENTERED FOR EVERY SEGMENT BEING ANALYZED.

= 1 TO 6; Σn TABLES 1 TO 6.

= 7; GENERAL EQUATION, $\Sigma n = (\Sigma N_{0i} e^{-(\Delta y)^2/2(\sigma_{\Delta y})_i^2})T$, i = 1, 2, 3

= 8; GUST EQUATION, $\Sigma n = (\Sigma N_{0i} P_i e^{-\Delta y/b_i \bar{A}})T$, i = 1, 2; $\bar{A}$ IS CALCULATED BY THE PROGRAM FOR AIRPLANE C.G. VERTICAL GUST LOAD FACTOR.

= 9; GUST EQUATION AS ABOVE; $\bar{A}$ IS DIRECTLY INPUT.

= 10 TO 12; (S_{MAX}, S_{MIN}, n) TABLES 1 TO 3.

= 13; GUST EQUATION AS ABOVE; K_{σ_u} AND $\bar{A}$ ARE CALCULATED BY THE PROGRAM FOR AIRPLANE C.G. VERTICAL GUST LOAD FACTOR.

= 14; GUST EQUATION AS ABOVE; $\bar{A}$ IS CALCULATED BY THE PROGRAM FOR AIRPLANE C.G. SIDE GUST LOAD FACTOR.

= 15; GUST EQUATION AS ABOVE; $\bar{A}$ IS CALCULATED BY THE PROGRAM FOR VERTICAL TAIL SIDE LOAD.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-6.2
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date____________

69	70 71	73	77 80
R R	CASE		1 6 P A
			PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
m or a_t					
Segm 1	656				
2	657				
3	658				
4	659				
5	660				
6	661				
7	662				
8	663				
9	664				
10	665				
11	666				
12	667				
13	668				
14	669				
15	670				
16	671				
17	672				
18	673				
19	674				
20	675				

QUAN	LOC	±	VALUE	±	E
m or a_t					
Segm 21	676				
22	677				
23	678				
24	679				
25	680				
26	681				
27	682				
28	683				
29	684				
30	685				
31	686				
32	687				
33	688				
34	689				
35	690				
36	691				
37	692				
38	693				
39	694				
40	695				

m = WING LIFT CURVE SLOPE PER RADIAN.
A VALUE MUST BE ENTERED ONLY FOR THE SEGMENTS WITH M3 = 8, OR 13.

a_t = LIFT CURVE SLOPE OF VERTICAL TAIL, PER RADIAN.
A VALUE MUST BE ENTERED ONLY FOR SEGMENTS WITH M3 = 14 OR 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III-6.6
STANDARD DATA INPUT 1

Page____ of______
Prepared by______
Date ______

69	70 71	73	77	80
R	R	CASE	1 6 P A	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
L or l_t					
Segm 1	3638				
2	3639				
3	3640				
4	3641				
5	3642				
6	3643				
7	3644				
8	3645				
9	3646				
10	3647				
11	3648				
12	3649				
13	3650				
14	3651				
15	3652				
16	3653				
17	3654				
18	3655				
19	3656				
20	3657				

QUAN	LOC	±	VALUE	±	E
L or l_t					
Segm 21	3658				
22	3659				
23	3660				
24	3661				
25	3662				
26	3663				
27	3664				
28	3665				
29	3666				
30	3667				
31	3668				
32	3669				
33	3670				
34	3671				
35	3672				
36	3673				
37	3674				
38	3675				
39	3676				
40	3677				

L = SCALE OF TURBULENCE, FT.
A VALUE MUST BE ENTERED ONLY FOR THOSE SEGMENTS WITH M3 = 13.

l_t = DISTANCE FROM AIRPLANE C.G. TO LIFT CENTER OF VERTICAL SURFACE, INCHES.
A VALUE MUST BE ENTERED ONLY FOR THOSE SEGMENTS WITH M3 = 14 or 15.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III - 6.7
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date______

69 70	71 73	77 80
		1 6 P A
R R	CASE	PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	±	VALUE	±	E
I_{yaw}					
Segm 1	3839				
2	3840				
3	3841				
4	3842				
5	3843				
6	3844				
7	3845				
8	3846				
9	3847				
10	3848				
11	3849				
12	3850				
13	3851				
14	3852				
15	3853				
16	3854				
17	3855				
18	3856				
19	3857				
20	3858				
$\overline{C}_t$	3837				
S_t	3838				

QUAN	LOC	±	VALUE	±	E
I_{yaw}					
Segm 21	3859				
22	3860				
23	3861				
24	3862				
25	3863				
26	3864				
27	3865				
28	3866				
29	3867				
30	3868				
31	3869				
32	3870				
33	3871				
34	3872				
35	3873				
36	3874				
37	3875				
38	3876				
39	3877				
40	3878				

I_{yaw} = MOMENT OF INERTIA ABOUT YAW AXIS, LB-IN^2
A VALUE MUST BE ENTERED ONLY FOR THOSE SEGMENTS WITH M3 = 14 OR 15.

$\overline{C}_t$ = MEAN GEOMETRIC CHORD OF VERTICAL SURFACE, FEET. ENTER ONLY IF M3 = 14 OR 15 IN ANY SEGMENT.

S_t = AREA OF VERTICAL TAIL, FT^2. ENTER ONLY IF M3 = 14 OR 15 IN ANY SEGMENT.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III -6.8
STANDARD DATA INPUT 1

Page____of______
Prepared by______
Date__________

69 70 (R R)	71 73 (CASE)	77 80 (PROG)
		16PA

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E	QUAN	LOC	± VALUE	± E
$C_{y_{\beta T}}$							
Segm 1	3879			Segm 21	3899		
2	3880			22	3900		
3	3881			23	3901		
4	3882			24	3902		
5	3883			25	3903		
6	3884			26	3904		
7	3885			27	3905		
8	3886			28	3906		
9	3887			29	3907		
10	3888			30	3908		
11	3889			31	3909		
12	3890			32	3910		
13	3891			33	3911		
14	3892			34	3912		
15	3893			35	3913		
16	3894			36	3914		
17	3895			37	3915		
18	3896			38	3916		
19	3897			39	3917		
20	3898			40	3918		

$C_{y_{\beta T}}$ = SIDE FORCE COEFFICIENT, TAIL CONTRIBUTION, PER RADIAN.

A VALUE MUST BE ENTERED ONLY FOR SEGMENTS WITH M3 = 14.

FATIGUE DAMAGE CALCULATION PROGRAM 16PA
FORTRAN DATA LOAD SHEET III - 6.9
STANDARD DATA INPUT 1

Page____ of____
Prepared by____
Date____

69 70	71 73	77 80
R R	CASE	16PA
		PROG

KEYPUNCH: DO NOT PUNCH BLANK DATA FIELDS.

QUAN	LOC	± VALUE	± E	QUAN	LOC	± VALUE	± E
$C_{y_{\beta to}}$							
Segm 1	3919			Segm 21	3939		
2	3920			22	3940		
3	3921			23	3941		
4	3922			24	3942		
5	3923			25	3943		
6	3924			26	3944		
7	3925			27	3945		
8	3926			28	3946		
9	3927			29	3947		
10	3928			30	3948		
11	3929			31	3949		
12	3930			32	3950		
13	3931			33	3951		
14	3932			34	3952		
15	3933			35	3953		
16	3934			36	3954		
17	3935			37	3955		
18	3936			38	3956		
19	3937			39	3957		
20	3938			40	3958		

$C_{y_{\beta to}}$ = Airplane side force coefficient, tail off, per radian. A value must be entered only for segments with M3 = 14.

APPENDIX B

PROGRAM A6PD ALTERNATE OPERATION

To generate a spectrum loading sequence through program A6PD requires a unique set of A6PA and A6PD input data, see Section V. Section IX describes a method of entering this data for each individual spectrum through a keypunched card deck. However, if one desires to generate a large number of spectrum variations from a basic input data set, an alternate method of input data management, which is more convenient and flexible, is available through the use of the CDC utility, UPDATE. This appendix describes this approach.

The CDC utility UPDATE allows the storing of the input data on one permanent file (disk). The data is to be divided into separate groupings, henceforth referred to as "decks". Through the use of UPDATE , any number of these decks can be retrieved to create a proper set of input data to generate one spectrum.

Each group of data is given a deck name. A *DECK name card must be placed at the beginning of each data group before the entire set of data is input to UPDATE to create a permanent data file (NEWPL). When retrieving these decks, it is necessary to place the deck names on compile directive cards (*COMPILE) which are input to UPDATE . Use as many compile cards as necessary and separate the deck names with commas:

*COMPILE name 1, name 2, name 5, etc.

For program A6PD, in the first *COMPILE card only, precede the first deck name entry with TWOCC3. This states that no word description is used in A6PA data (data sheet I-1 in Reference 1) and no such data should be entered into UPDATE . The data will be read into program A6PD in the sequence of deck names listed on *COMPILE cards. Consequently, first list the A6PA data and then the A6PD data. For further information on the CDC utility UPDATE see Reference 4.

One method of dividing the program A6PD input data for storing on the UPDATE file, used in Reference 2 to generate a large number of spectrum variations, is:

A6PA data: Reference Run (Case 000) data

Case data

A6PD data.

Do not include the last A6PA data card (1 in cc1 and 99999 in cc69-73) with the A6PA data but include as the first card with the A6PD data.

To illustrate the use of UPDATE to generate a spectrum loading sequence, the test case of Section X from Reference 2, will be used. In Reference 2, over one hundred spectra variations were generated from a basic set of data and additional data to produce the variations. The following deck names were given to the individual groups of data:

A6PA data:

Basic (original):	RRXX ,	Ref. Run = XX, Case 000
	CXXYYY ,	Ref. Run = XX, Case = YYY
Variation:	RRXX/SZZ,	Ref. Run = XX, Case 000, Spectrum = ZZ
	CXXY/SZZ,	Ref. Run = XX, Case = Y, Spectrum = ZZ
A6PD data:	SZZ ,	Spectrum = ZZ

Each spectrum was given an identification number ZZ . However, if applicable, data deck of any spectrum can be used to generate any other spectrum. It must be remembered that the A6PA data is read into the program by a special subroutine INPUT1 which has the feature of replacing previously read data with a later entry. The deck setup (list of control cards) required to generate the test case (spectrum 1 or BS1 in Reference 2) using Douglas Aircraft Company CDC computer, running under KRONOS, is given on the following page.

Job Control Cards to Generate Test Case Spectrum (Spectrum #1, BS1) Using CDC UPDATE Utility

```
SER,AL01.
BS1,MFL155000,T___,IO___,L__.
ACCOUNT, , .
CID, Name, etc.
REQUEST,TAPE3,NT,LB=KU,D=1600,VSN=RESERVE,T____.
ATTACH,LGO=A6PDLGO,etc.
ATTACH,OLDPL=A6DATA6,etc.                    Note 1.
UPDATE,Q,D,8,C=DATA.
COPYSBF,DATA,OUTPUT.                         Note 2.
REWIND,DATA.
LGO,DATA.
/EOR
*COMPILE TWOCC3,RR15,C15001,C151/S1,C15002
*COMPILE C152/S1,S1
/EOF
```

Note 1. A6DATA6 is the complete UPDATE data filename.

Note 2. Use this card only if you wish to get a list of the input data (card images) for the spectrum being run. The list will appear in the printout before the program output.

☆U.S.Government Printing Office: 1979 — 657-002/508